Nonparametric Learning for Layered Segmentation of Natural Images

Soumya Ghosh and Erik B. Sudderth
Department of Computer Science, Brown University, Providence, RI, USA
sghosh@cs.brown.edu, sudderth@cs.brown.edu

Abstract

We explore recently proposed Bayesian nonparametric models of image partitions, based on spatially dependent Pitman-Yor processes. These models are attractive because they adapt to images of varying complexity, successfully modeling uncertainty in the structure and scale of human segmentations of natural scenes. By developing substantially improved inference and learning algorithms, we achieve performance comparable to state-of-the-art methods. For learning, we show how the Gaussian process (GP) covariance functions underlying these models can be calibrated to accurately match the statistics of example human segmentations. For inference, we develop a stochastic search-based algorithm which is substantially less susceptible to local optima than conventional variational methods. Our approach utilizes the expectation propagation algorithm to approximately marginalize latent GPs, and a low rank covariance representation to improve computational efficiency. Experiments with two benchmark datasets show that our learning and inference innovations substantially improve segmentation accuracy. By hypothesizing multiple partitions for each image, we also take steps towards capturing the variability of human scene interpretations.

1. Introduction

Image segmentation algorithms partition images into spatially coherent, approximately homogeneous regions. Segmentations provide an important mid-level representation which can be leveraged for various vision tasks including object recognition [11], motion estimation [26], and image retrieval [4]. Despite significant research [23, 5, 7, 15, 2], segmentation remains a largely unsolved problem. One major challenge is to move beyond seeking a single "optimal" image partition, and to recognize that while there are commonalities among multiple human segmentations of the same image, there is also substantial variability [12].

Most existing segmentation algorithms are endowed with a host of tunable parameters; a particular configuration may work well on some images, and poorly on others. Often these parameters are tuned via manual experimentation, or expensive validation experiments. Noting this issue, Russell et al. [21] produced a "soup of segments" by varying the parameters of the normalized cuts algorithm, and collecting the range of observed outputs. Others have used agglomerative clustering methods to produce a nested tree of segmentations [2]. A limitation of these approaches is that they do not provide any image-specific estimate of which particular segmentations are most accurate.

In this paper, we instead pursue a Bayesian nonparametric statistical approach to modeling segmentation uncertainty. We reason about prior and posterior distributions on the space of image partitions, and thus consider segmentations of all possible resolutions. In contrast with parametric segmentation models based on finite mixtures [4, 1, 22] or Markov random fields [8], we do *not* need to pre-specify the number of segments. Our inference algorithm automatically provides calibrated estimates of the relative probabilities of segmentations with varying numbers of regions.

Because we define a consistent probabilistic model and not just a segmentation procedure, our approach is a natural building block for more sophisticated models. We improve earlier work on spatially dependent Pitman-Yor (PY) processes [25], which was motivated by the problem of jointly segmenting multiple related images. This PY model was later extended to allow prediction of semantic segment labels, given supervised annotations of objects in training images [24]. Here we focus on the problem of segmenting single images containing unknown object categories.

The model we consider is a minor variation on the dependent PY process of Sudderth and Jordan [25], which captures the power law distribution of human image segments via a stick-breaking construction, and uses Gaussian processes (GPs) to induce spatial dependence. Our first major contribution is a new posterior inference algorithm that is far less susceptible to local optima than previous mean field variational methods [25]. Our algorithm combines a discrete stochastic search, capable of making large moves in the space of image partitions, with an accurate higher-order variational approximation (based on expectation propagation [14]) to marginalize latent GPs. We improve computational efficiency via a low rank representation of the GP covariance, an innovation that could be applicable to many

other models with high-dimensional Gaussian variables.

Our second major contribution is a procedure for learning the various model hyperparameters, including image-dependent GP covariance functions, from example human segmentations. Using training images from the Berkeley segmentation dataset [12], we calibrate our model, and then evaluate its accuracy in segmenting various images of natural scenes [12, 16]. Our results show significant improvements over prior work with PY process models [25], and demonstrate segmentations that are both qualitatively and quantitatively competitive with state-of-the-art methods.

2. Nonparametric Bayesian Segmentation

We have two primary requirements of any segmentation model – a) it should adapt to image complexity and automatically select the appropriate number of segments and b) it should encourage spatial neighbors to cluster together. Furthermore, human segmentations of natural scenes consist of segments of widely varying sizes. It has been observed that histograms over segment areas [12] and contour lengths [19] are well explained by power law distributions. Thus a third requirement is to model this power-law behavior. In this section, we first describe our image representation and then review increasingly sophisticated models which satisfy these requirements. Finally, in Sec. 2.4, we propose a novel low-rank model which improves computational efficiency while retaining the above desiderata .

2.1. Image Representation

Each image is dicided into roughly 1,000 *superpixels* [20] using the normalized cuts spectral clustering algorithm [23]. The color of each superpixel is described using a histogram of HSV color values with $W_c = 120$ bins. We choose a non-regular quantization to more coarsely group low saturation values. Similarly, the texture of each superpixel is modeled via a local $W_t = 128$ bin texton histogram [13], using quantized band-pass filter responses. Superpixel n is then represented by histograms $x_n = (x_n^t, x_n^c)$ indicating its texture x_n^t and color x_n^c.

2.2. Pitman-Yor Mixture Models

Pitman-Yor mixture models extend traditional finite mixture models by defining a Pitman-Yor (PY) process [17] prior over the distribution of mixture components. The distributions sampled from a PY process are countably infinite discrete distributions which place mass on infinitely many mixture components. Furthermore, these discrete distributions follow a power law distribution and previous work [25] has shown that they model the distribution over human segment sizes well. There are various ways of formally defining the PY process, here we consider the stick breaking representation. Let $\boldsymbol{\pi} = (\pi_1, \pi_2, \pi_3, \ldots)$, $\sum_{k=1}^{\infty} \pi_k = 1$, denote an infinite *partition* of a unit area region (in our case, an image). The Pitman-Yor process defines a prior distribution on this partition via the following *stick-breaking* construction:

$$\pi_k = w_k \prod_{\ell=1}^{k-1} (1 - w_\ell) = w_k \Big(1 - \sum_{\ell=1}^{k-1} \pi_\ell \Big) \qquad (1)$$
$$w_k \sim \mathrm{Beta}(1 - \alpha_a, \alpha_b + k\alpha_a)$$

This distribution, denoted by $\boldsymbol{\pi} \sim \mathrm{GEM}(\alpha_a, \alpha_b)$, is defined by two hyperparameters (the discount and the concentration parameters) satisfying $0 \leq \alpha_a < 1$, $\alpha_b > -\alpha_a$. It can be shown that $\mathbb{E}[\pi_k] \propto k^{-1/\alpha_a}$, thus exhibiting the aforementioned power law distribution.

For image segmentation, each index k is associated with a different segment or region with its own appearance models $\theta_k = (\theta_k^t, \theta_k^c)$ parameterized by multinomial distributions on the W_t texture and W_c color bins, respectively. Each superpixel n then independently selects a region $z_n \sim \mathrm{Mult}(\boldsymbol{\pi})$, and a set of quantized color and texture responses according to

$$p(x_n^t, x_n^c \mid z_n, \boldsymbol{\theta}) = \mathrm{Mult}(x_n^t \mid \theta_{z_n}^t, M_n)\, \mathrm{Mult}(x_n^c \mid \theta_{z_n}^c, M_n) \qquad (2)$$

The multinomial distributions themselves are drawn from a symmetric Dirichlet prior with hyper-paramter ρ. Note that conditioned on the region assignment z_n, the color and texture features for each of the M_n pixels within superpixel n are sampled independently. The appearance feature channels provide weak cues for grouping superpixels into regions. Since, the model doesn't enforce any spatial neighborhood cues, we refer to it as the "bag of features" (*BOF*) model.

2.3. Spatially Dependent PY Mixtures

Next, we review the approach of Sudderth and Jordan [25] which extends the BOF model with spatial grouping cues. The model combines the BOF model with ideas from layered models of image sequences [28], and level set representations for segment boundaries [6].

We begin by elucidating the analogy between PY processes and layered image models. Consider the PY stick-breaking representation of Eq. (1). If we sample a random variable z_n such that $z_n \sim \mathrm{Mult}(\boldsymbol{\pi})$ where $\pi_k = w_k \prod_{\ell=1}^{k-1}(1 - w_\ell)$, it immediately follows that $w_k = \mathbb{P}[z_n = k \mid z_n \neq k-1, \ldots, 1]$. The stick-breaking proportion w_k is thus the *conditional* probability of choosing segment k, given that segments with indexes $\ell < k$ have been rejected. If we further interpret the ordered PY segments $\{k = 1, \ldots \infty\}$ as a sequence of layers, z_n can be sampled by proceeding through the layers in order, flipping biased coins (with probabilities w_k) until a layer is chosen. Given this, the probability of assignment to subsequent layers is zero; they are effectively *occluded* by the chosen "foreground" layer.

The spatially dependent Pitman-Yor process of [25] pre-

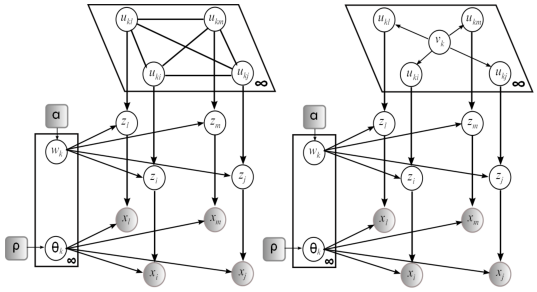

Figure 1. **Generative models of image partitions.** *Left*. Spatially dependent PY model, *(right)* low rank model. Shaded nodes represent observed random variables. $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_D)$ is a low dimensional Gaussian random variable and $\mathbf{u}_k$ is the corresponding N dimensional layer. $w_k \sim \text{Beta}(1-\alpha_a, \alpha_b + k\alpha_a)$ controls expected layer size and are governed by Pitman-Yor hyper-parameters $\alpha = (\alpha_a, \alpha_b)$. The Dirichlet hyper-parameters $\rho = (\rho^t, \rho^c)$ parametrize appearance distributions. Finally, the color and texture histograms describing super-pixel n are represented as $x_n = (x_n^t, x_n^c)$

serves this PY construction, while adding spatial dependence among super-pixels by associating a layer (real valued function) drawn from a zero mean *Gaussian process* (GP) $\mathbf{u}_k \sim GP(\mathbf{0}, \Sigma)$ with each segment k. Σ captures the spatial correlation amongst super-pixels, and without loss of generality we assume that it has a unit diagonal. Each super-pixel can now be associated with a layer following the procedure described in the previous paragraph, n.e.,

$$z_n = \min\left\{k \mid u_{kn} < \Phi^{-1}(w_k)\right\}, \; u_{kn} \sim \mathcal{N}(0, \Sigma_{nn} = 1) \quad (3)$$

Here, $u_{kn} \perp u_{\ell n}$ for $k \neq \ell$ and $\Phi(u)$ is the standard normal *cumulative distribution function* (CDF). Let $\delta_k = \Phi^{-1}(w_k)$ denote a threshold for layer k. Since $\Phi(u_{kn})$ is uniformly distributed on $[0, 1]$, we have

$$\begin{aligned} \mathbb{P}[z_n = 1] &= \mathbb{P}[u_{1n} < \delta_1] = \mathbb{P}[\Phi(u_{1n}) < w_1] = w_1 = \pi_1 \\ \mathbb{P}[z_n = 2] &= \mathbb{P}[u_{1n} > \delta_1]\, \mathbb{P}[u_{2n} < \delta_2] = (1 - w_1) w_2 = \pi_2 \end{aligned} \quad (4)$$

and so on. The extent of each layer is determined via the region on which a real-valued function lies below the threshold δ_{layer}, akin to level set methods. If $\Sigma = \mathbf{I}$, we recover the BOF model. More general covariances can be used to encode the prior probability that each feature pair occupies the same segment; developing methods for learning these probabilities is a major contribution of this paper.

The power law prior on segment sizes is retained by transforming priors on stick proportions $w_k \sim \text{Beta}(1-\alpha_a, \alpha_b + k\alpha_a)$ into corresponding randomly distributed thresholds $\delta_k = \Phi^{-1}(w_k)$:

$$p(\delta_k \mid \alpha) = \mathcal{N}(\delta_k \mid 0, 1) \cdot \text{Beta}(\Phi(\delta_k) \mid 1 - \alpha_a, \alpha_b + k\alpha_a) \quad (5)$$

Figure 1 displays corresponding graphical model. Image features are generated as in the BOF model.

2.4. Low-Rank Representation

In the preceding generative model, the layer support functions $\mathbf{u}_k \sim \mathcal{N}(0, \Sigma)$ are samples from a Gaussian distribution over N super-pixels. Inference involving GPs involve inverting Σ which is in general a $O(N^3)$ operation and thus scales poorly with increasing image sizes. To cope, we employ a low-rank representation based on $D \leq N$ dimensions, analogous to factor analysis models. We proceed by defining a Gaussian distributed D dimensional latent variable $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_D)$, we then set $\mathbf{u}_k = A\mathbf{v}_k + \epsilon_k$, where A is a N-by-D dimensional factor loading matrix and $\epsilon_k \sim \mathcal{N}(0, \Psi)$, with Ψ being a diagonal matrix. Observe that marginalizing over $\mathbf{v}_k$ results in a model equivalent to the full rank model of the preceding section with $\Sigma = AA^T + \Psi$. The low rank model replaces the $O(N^3)$ operation with an $O(ND^2)$ operation, thus scaling linearly with N[1]. Figure 1 displays the corresponding graphical model.

3. Inference

This section describes a novel, robust to local optima, inference algorithm which is an example of a Maximization Expectation (ME) [29] technique. In contrast to the popular Expectation Maximization algorithms, ME algorithms marginalize model parameters and directly maximize over the latent variables. In our model, the latent variables correspond to segment assignments of super-pixels (z_n). Any configuration of these variables defines a partition of the image. Our strategy is to explore the space of these image partitions by climbing the posterior $p\left(\mathbf{z} \mid \mathbf{x}, \eta\right)$ surface, where $\eta = \{\alpha, \rho, A, \Psi\}$. It is worth noting that since different partitions will have different numbers of segments, we are in fact searching over models of varying complexities akin to traditional model selection techniques.

The algorithm proceeds by first evaluating the posterior for an initial image partition $\mathbf{z}$. It then modifies the partition in an interesting fashion to generate a new partition $\mathbf{z}'$ which is accepted if $p(\mathbf{z}' \mid \mathbf{x}, \eta) \geq p(\mathbf{z} \mid \mathbf{x}, \eta)$. This process is repeated until convergence. By caching the various mutated partitions, we approximate the posterior distribution over partitions (Figure 5). In what follows, we first describe the innovations required for evaluating the posterior marginal and then the procedure for mutating a partition.

3.1. Posterior Evaluation

In our model (*Figure 1*), the posterior $p\left(\mathbf{z} \mid \mathbf{x}, \eta\right)$ factorizes as $p\left(\mathbf{z} \mid \mathbf{x}, \eta\right) \propto p\left(\mathbf{x} \mid \mathbf{z}, \rho\right) p\left(\mathbf{z} \mid \alpha, A, \Psi\right)$. The likelihood:

$$p\left(\mathbf{x} \mid \mathbf{z}, \rho\right) = \int_{\Theta} p\left(\mathbf{x} \mid \mathbf{z}, \Theta\right) p\left(\Theta \mid \rho\right) d\Theta \quad (6)$$

[1] A complete time complexity analysis is available in the supplement.

is a standard Dirichlet-multinomial integral and can be evaluated in closed form[2].

Unfortunately, the prior can't similarly be evaluated in closed form. Significant innovations are required for its computation and the remainder of this section details a major contribution of this paper, an algorithm for evaluating $p\left(\mathbf{z} \mid \eta\right)$.

$$p\left(\mathbf{z} \mid \eta\right) = \prod_{k=1}^{K(\mathbf{z})} \int_{\mathbf{u}_k} \int_{\delta_k} \int_{\mathbf{v}_k} p\left(\mathbf{z} \mid \delta_k, \mathbf{u}_k\right) p\left(\mathbf{u}_k, \mathbf{v}_k \mid A, \Psi\right) p\left(\delta_k \mid \alpha\right) d\mathbf{v}_k d\mathbf{u}_k d\delta_k \tag{7}$$

where $K(\mathbf{z})$ represents the number of layers in partition $\mathbf{z}$. To simplify notation in the remainder of this paper we denote $K(\mathbf{z})$ simply by K. Note that in the BOF model $\mathbf{z}$ depends only on α and $p(\mathbf{z}|\alpha)$ can be calculated in closed form:

$$p(\mathbf{z} \mid \alpha) = \alpha_a^K \frac{\Gamma\left(\alpha_b/\alpha_a + K\right)\Gamma(\alpha_b)}{\Gamma(\alpha_b/\alpha_a)\Gamma(N+\alpha_a)} \left(\prod_{k=1}^{K} \frac{\Gamma(M_k - \alpha_a)}{\Gamma(1-\alpha_a)}\right) \tag{8}$$

where N is the number of super-pixels in the partition and M_k is the number of super-pixels in layer k.

Spatial prior evaluation. The integrals in equation 7 can be evaluated independently for each layer. In the following analysis, it is implied that we are dealing with the k^{th} layer and we drop the explicit dependence on k in our notation. We approximate the joint distribution $p(\mathbf{u}, \mathbf{v}, \delta, z \mid \eta)$ with a Gaussian distribution $q(\mathbf{u}, \mathbf{v}, \delta, \mathbf{z} \mid \eta)$ and the corresponding marginal $p(\mathbf{z} \mid \eta)$ with $q(\mathbf{z} \mid \eta)$, which is easy to compute. We use expectation propagation (EP) [14] to estimate the Gaussian "closest" to the true joint distribution.

Recall that our model assigns super-pixel n to the first layer k whose value is less than the layer's threshold (δ), thus setting $z_n = k$. Equivalently, we can introduce a binary random variable t_n for each layer k, whose value is deterministically related to z_n as follows:

$$t_n = \begin{cases} +1 & \text{if } z_n = k \implies u_n < \delta \\ -1 & \text{if } z_n > k \implies u_n > \delta \end{cases} \tag{9}$$

Note that super-pixels with $z_n < k$ have already been assigned to preceding layers and can be marginalized out before inferring the latent Gaussian layer for the k^{th} layer. We can now express the joint distribution in terms of $\mathbf{t}$:

$$p(\mathbf{u}, \mathbf{v}, \delta, \mathbf{t} \mid \eta) = p(\mathbf{v})\; p(\delta \mid \alpha) \prod_{n=1}^{N} p(u_n \mid \mathbf{v}) p(t_n \mid u_n, \delta) \tag{10}$$

Furthermore, since for a given partition $\mathbf{t}$ is known, we can condition on it to get

$$p(\mathbf{u}, \mathbf{v}, \delta \mid \mathbf{t}, \eta) = \frac{1}{Z}\mathcal{N}(\mathbf{v} \mid 0, I)\; p(\delta|\alpha) \prod_{n=1}^{N} \mathcal{N}(u_n \mid a_n^T \mathbf{v}, \psi_n)\mathbb{I}(t_n(\delta - u_n) > 0) \tag{11}$$

where Z is the appropriate normalization constant. Note that the indicator functions $\mathbb{I}(t_n(\delta - u_n) > 0)$ and the threshold prior $p(\delta \mid \alpha)$ are the only non Gaussian terms. We approximate these with un-normalized Gaussians, leading to the following approximate posterior

$$q(\mathbf{u}, \mathbf{v}, \delta \mid \mathbf{t}, \eta) = \frac{1}{Z_{\text{EP}}}\mathcal{N}([\mathbf{v}^T\; \mathbf{u}^T\; \delta]^T \mid \mu_{\approx}, \mathbf{\Sigma}_{\approx}) \tag{12}$$

where Z_{EP} ensures appropriate normalization. We now iteratively refine the Gaussian approximation using EP [3]. At convergence we compute $Z_{\text{EP}} = \int_{\mathbf{u}} \int_{\mathbf{v}} \int_{\delta} q(\mathbf{u}, \mathbf{v}, \delta, \mathbf{t} \mid \eta)$ which is prior for the k^{th} layer. Finally, we have $p(\mathbf{z}|\eta) \approx \prod_{k=1}^{K} Z_{\text{EP}_k}$.

With the expression for prior in hand, we can now compute the log posterior marginal

$$\log\; p(\mathbf{z} \mid \mathbf{x}, \eta) = \gamma\; \log\; p(\mathbf{x} \mid \mathbf{z}, \rho) + \sum_{k=1}^{K} \log Z_{\text{EP}_k} \tag{13}$$

The parameter γ is used to weight the likelihood appropriately. We set $\gamma = \frac{1}{\bar{m}}$, where $\bar{m}$ is the average number of pixels per super-pixel. Recall that our likelihood treats pixels within a super-pixel as independent random variables, necessitating the above down weighting.

3.2. Search over partitions

Armed with the ability to evaluate the posterior probability mass for a given image partition, we explore the space of partitions using discrete search. The search performs hill climbing on the posterior surface and explores high probability regions of the partition space. This is similar in spirit to MCMC techniques. Perhaps most similar to our approach is the data driven MCMC approach of Tu *et al*. [27], which uses a version of the Metropolis-Hastings algorithm along with clever data driven proposals to explore the posterior space. Here, we forgo the requirement of *eventually* converging to the true posterior distribution in exchange for the ease of incorporating flexible search moves and the ability to quickly explore high probability regions of the posterior. Given a partition we propose a new candidate partition by stochastically choosing one of the following moves:

Merge. Two layers in the current partition are merged into a single layer.

Split. A layer is split into two layers, which are adjacent in layer order. We employ two types of shift moves. Given a

[2]The result follows from Dirichlet multinomial conjugacy. Please see the supplement for relevant details

[3]Applying EP to our low dimensional model requires an interesting combination of Gaussian belief propagation and expectation propagation. Due to space limitations we haven't included the details of EP here, but all relevant details can be found in the supplement.

layer to be split, the first move works by randomly selecting two seed super-pixels and then assigning all remaining super-pixels to the closest (in appearance space) seed. The initial seeds are chosen such that with high probability they are far in appearance space. The second move employs a connected component operation. If the given layer has disconnected components then one such disconnected component is sampled at random and deemed to be a new layer.

Swap. The swap move reorders the layers in the current partition, by selecting two layers and exchanging their order.

Shift. The shift move refines the partitions found by the other moves. It iterates over all super-pixels in the image assigning each to a segment which maximizes the posterior probability [4]. Observe that the merge and split moves change the number of layers in a partition performing model selection, while swap and shift attempt to find the optimal partition given a model order.

4. Learning from Human Segmentations

In this section, we provide methods for quantitatively calibrating the proposed models to appropriate human segmentation biases. Recall that our model has four hyper-parameters, the PY region size hyper-parameter (α), the appearance hyper-parameter (ρ) and the GP covariance parameters (A and Ψ). We tune these to the human segmentations from the 200 training images of the Berkeley Segmentation Dataset (BSDS) [12]. We show that in spite of the inherent uncertainty in the segmentations of an image, we are able to learn important low level grouping cues.

Learning size and appearance hyper-parameters. The optimal region size hyper-parameters are the ones that best describe the statistics of the training data. We select $\hat{\alpha} = (\hat{\alpha}_a, \hat{\alpha}_b)$ by performing a grid search over 20 evenly spaced α_a and α_b candidates in the intervals $[0, 1]$ and $[0.5, 20]$ respectively and choosing values which maximize the model's likelihood of the training partitions according to equation 8. The appearance hyper-parameters $\hat{\rho} = (\hat{\rho}^t, \hat{\rho}^c)$ are tuned through cross validation on a subset of the training set. For BSDS, the estimated parameters equal $\hat{\alpha}_a = 0.15, \hat{\alpha}_b = 1$ $\hat{\rho}^t = 0.01$ and $\hat{\rho}^c = 0.01$

Learning covariance kernel hyper-parameters. The covariance kernel governs the type of layers that can be expressed by the model. Estimating it accurately is crucial for accurately partitioning images. In [25, 24] the authors use various heuristics to specify this kernel. Here, we take a more data driven approach and learn the kernel from human segmentations. While we cannot expect our training data to provide examples of all important region appearance patterns, it does provide important cues. In particular like [9], we learn to predict the probability that *pairs* of super-pixels occupy the same segment via human segmentations.

For every pair of super-pixels, we consider several potentially informative low-level cues: (i) pairwise Euclidean distance between super-pixel centers; (ii) intervening contours, quantified as the maximal response of the probability of boundary (Pb) detector [13] on the straight line linking super-pixel centers; (iii) local feature differences, estimated via log empirical likelihood ratios of χ^2 distances between super-pixel color and texture histograms [20]. To model non-linear relationships between these four raw features and super-pixel groupings, each feature is represented via the activation of 20 radial basis functions, with the appropriate bandwidth chosen by cross-validation. Concatenating these gives a feature vector ϕ_{ij} for every super-pixel pair i, j. We then train a L_2 regularized logistic regression model to predict the probability of two super-pixels occupying the same segment q_{ij}. Figure 2 illustrates the effect of these cues on partitions preferred by the model.

When probabilities are chosen to depend only on the distance between super-pixels the distribution constructed defines a generative model of image features. When these probabilities also incorporate contour cues, the model becomes a conditionally specified distribution on image partitions, analogous to a conditional random field [10].

From probabilities to correlations. Recall that our layers are functions sampled from multivariate Gaussian distributions, with covariance Σ with unit variance and a potentially different correlation c_{ij} for each super-pixel pair i, j. For each super-pixel pair, q_{ij} is *independently* determined by the corresponding correlation coefficient c_{ij}. As detailed in the supplement there exists an one-to-one mapping between the pairwise probabilities and correlations, allowing us to go from the logistic regression outputs (q_{ij}) to correlation matrices. These correlation matrices (C), learned from pairwise probabilities will in general not be positive semi-definite (PSD). We cope by finding the closest PSD unit diagonal matrix to the correlation matrix. We use the recently proposed technique of Borsdorf *et al.* [3], which solves for A and Ψ by minimizing the Frobenius norm$||C - (AA^T + \Psi)||_F$. It should be noted that even the heuristic approaches of Sudderth and Jordan [25] and Shyr *et al.* [24] can yield non PSD correlation matrices. There the authors ensure positive semi-definiteness by performing an eigen-decomposition of C and retaining only non-negative eigenvalues. This is a cruder approximation and leads to poor results (Figure 2).

5. Spatially dependent PY model properties

In this section, we explore various properties of our model which may not be immediately obvious.

[4] A naive shift move would evaluate the posterior probability of the partition after every super-pixel shift. This proves to be prohibitively expensive, instead we develop an alternative which allows us to evaluate the posterior after one complete sweep through the super-pixels while ensuring that each individual shift by-and-large increases the posterior. Please see the supplement for details.

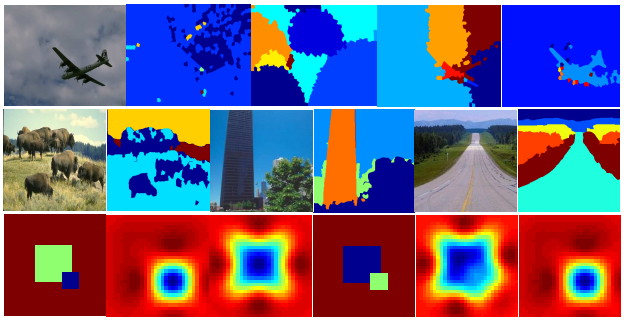

Figure 2. **Model Properties**. *TOP*- Prior samples from models employing heuristic distance+pb [25], learned distance (PYdist) , learned distance+pb and all cues (PYall) based covariances. *CENTER*- Layered segmentations produced by our method. *BOTTOM* - Three layer synthetic partitions illustrating preferred layer orderings, Layer 1 is displayed in blue and Layer 2 in green. *Left to right:* Partition 1 (*blue = low; red = high*), the inferred Gaussian function for layers 1 and 2, partition 2 and the corresponding Gaussian functions. Under our model, partition 1 has a log probability of -77 while partition 2 has a log probability of -90.

Prior samples. Our model defines a distribution over image partitions, which can be partially assessed by visualizing partitions sampled from the prior. Figure 2 displays such samples. Note that the samples from the conditionally specified models better reflect the structure of the image.

Layers. Our model produces partitions made up of layers, not segments. These layers can have multiple connected components, due to either occlusion by a foreground layer, or a layer support function with multimodal shape. The inferred partitions illustrated in the second row of figure 2 illustrate this point. The model groups all buffaloes (in the first image), non-contiguous portions of sky, grass and trees (in the second and third images) in the same layer. Traditional segmentation algorithms, having no notion of layers, would assign each non contiguous region to a separate segment. Our layered representation provides a higher level representation of the scene than is possible with a collection of segments, which allows us to naturally deal with complex visual phenomena such as occlusion.

Implicit prior on layer order. Recall that a partition is an ordered sequence of layers, and the likelihood of a partition is governed by the likelihood of its constituent layers. Note that reordering layers can change the set of support functions which produce those layers, which in turn makes certain orderings preferable to others. In general, our GP priors prefer simple shapes over complicated ones and hence our model prefers explaining complicated shapes via an occlusion process. Figure 2 illustrates these ideas using two synthetic partitions with the order of layers 1 and 2 flipped. The model [5] prefers the partition in the first column over the one in fourth. As can be seen from the inferred layers, partition 1 is explained by the model using simpler Gaussian functions, while partition 2 has to be explained using more complicated and hence less likely Gaussian functions.

[5]Here, we have used a squared exponential covariance kernel with length scale set to half of the partition's diagonal length.

6. Experimental Results

In this section we present quantitative evaluations of various aspects of the proposed model along with qualitative results. In all experiments, our model (PYall) used a 200 dimensional low rank representation and ran 200 discrete search iterations, with three random restarts.

Experimental Setup. We benchmark the algorithm on the Berkeley Image Segmentation Dataset (BSDS300 [12]) and a subset of of Oliva and Torralba's [16] eight natural categories dataset. We sampled the first 30 images from each of the eight categories to create a 240 image dataset.

The performance of the algorithms are quantified using the probabilistic Rand Index (*PRI*) [18], and the segmentation covering (*SegCover*) metric [2]. The partitions produced by our model are made up of layers, which may not be spatially contiguous. However, the benchmarks we evaluate on, define segments to be spatially contiguous regions. To produce these we run connected components on the layers splitting them into spatially contiguous segments.

Quantifying model enhancements. This paper improves on both the model (PYheur) and the corresponding inference algorithm presented in [25] . To quantify the performance gains solely from model enhancements we devise the following test. On BSDS300 test images, we compare the log-posterior assigned to the ground truth human segmentations $p(z_{gt}|x,\eta)$ under both models. Since, we already have access to z_{gt} no inference is required and the model which assigns higher probability mass to the ground truth, models the data better. Figure 3 presents a scatter plot comparing both models. It is easy to see that PYall models human segmentations significantly better.

Evaluating inference enhancements. Next, we evaluate the performance improvements resulting from the novel inference algorithm[6]. Figure 3 displays the result of running mean field and search based inference from 10 random initializations for a given test image. The log-likelihood plots clearly demonstrate mean field being susceptible to local minima. In contrast, EP based search exhibits robustness and all chains converge to high probability partitions. The bottom row displays the best and worst partitions found by mean field and search. As one would expect, there is wide variability in the quality of mean field partitions, while the search partitions are consistently good. The rightmost top row plot displays randomly chosen partitions from the 10 EP search runs. It demonstrates a high correlation between

[6]100 search iterations takes about 30 minutes on a standard quadcore with 4GB of ram.

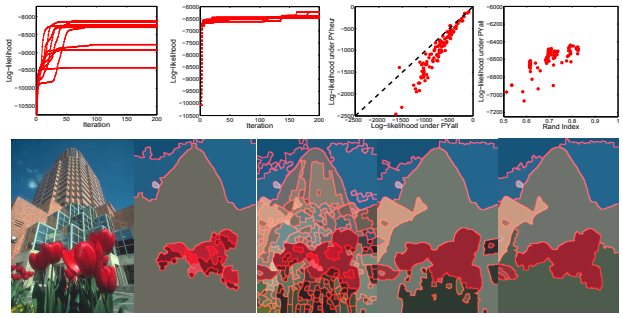

Figure 3. **Model and inference comparison**. *TOP (Left to right)* Log-likelihood (ll) trace plots of mean field runs, search runs, scatter plot comparing PYall and PYheur, scatter plot of ll vs Rand index. *BOTTOM (Left to right)* Test image, partitions with highest and lowest ll found by mean field, best and worst search partitions.

	BSDS300							LabelMe	
	Ncuts	MS	FH	gPb	PYheur	PYdist	PYall	gPb	PYall
PRI	0.73	0.77	0.77	0.80	0.60	0.69	0.76	0.74	0.73
segCover	0.40	0.48	0.53	0.58	0.45	0.50	0.54	0.54	0.55

Table 1. Quantitative performance of various algorithms on BSDS300 and LabelMe.

log likelihoods and rand indexes, again verifying that the partitions favored by our model are also favored by humans.

Comparison against competing methods. In this paper, our goal is not to produce one "optimal" segmentation but to provide a tractable handle on the posterior distribution over image partitions. Nevertheless, here we demonstrate that by summarizing the posterior with the MAP partition we produce results which are competitive with the state-of-the-art segmentation techniques. We compare against four popular segmentation techniques: Mean Shift (MS) [5], Felzenszwalb and Huttenocher's graph based segmentation (FH) [7], Normalized cuts [23] and gPb contour based segmentation [2][7]. In addition, we also compare against a version of our model which uses only distance cues for learning the covariance kernel (PYdist). Table 1 displays the quantitative numbers achieved on the BSDS300 test set. Figure 4 demonstrates qualitative differences amongst the methods. PYall is significantly better than both PYheur and PYdist. According to a Wilcoxon's signed rank test (at an 0.01 significance level) it is also significantly better than Ncuts and MS (on segCover metric, within noise on PRI), within noise of FH and statistically worse than gPb on the BSDS300 dataset.

Next, in order to test generalizability, we compare PYall against the top performing method on BSDS – gPb on the LabelMe dataset. The parameters for either method were tuned on BSDS and were not re-tuned to the LabelMe dataset. Table 1 displays the results. PYall and gPb are now statistically indistinguishable.

[7] All model parameters were tuned by performing a grid search on the training set. See supplement for more details.

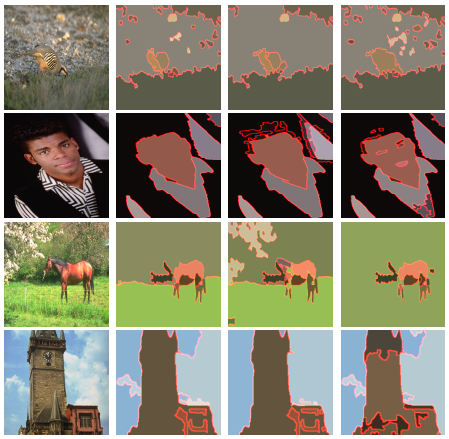

Figure 5. **Diverse Segmentations.** Each row depicts multiple partitions for a given image. Partitions in the second column are the MAP estimates. Other partitions with significant probability masses are shown in the third and fourth columns.

Posterior Summary. Perhaps, a more accurate assessment of our model involves exploring the posterior distribution over partitions. In Figure 5 we summarize the posterior distributions, for a few randomly chosen test images, by presenting a set of high probability partitions discovered by our algorithm. It is worth noting that the set of multiple partitions produced by our method is richer than those produced by a single multi-resolution segmentation tree [2]. For instance, the partitions in the third and fourth columns of the first two rows of Figure 5 are mutually inconsistent with any one segmentation tree, but are nonetheless produced by our algorithm. More interesting ways of leveraging the distribution over partitions is an important direction of future work.

7. Discussion

Starting with a promising Bayesian nonparametric model of images partitions, we have developed substantially improved algorithms for learning from example human segmentations, and robustly inferring multiple plausible segmentations of novel images. By defining a consistent distribution on segmentations of varying resolution, this dependent PY process provides a promising building block for other high-level vision tasks.

References

[1] M. Andreetto, L. Zelnik-Manor, and P. Perona. Non-parametric probabilistic image segmentation. In *ICCV*, 2007.

[2] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. From contours to regions: An empirical evaluation. *CVPR*, 0:2294–2301, 2009.

[3] R. Borsdorf, N. J. Higham, and M. Raydan. Computing a nearest correlation matrix with factor structure. *SIAM J. Matrix Analysis App.*, 31(5):2603–2622, 2010.

[4] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *PAMI*, 24(8):1026–1038, Aug. 2002.

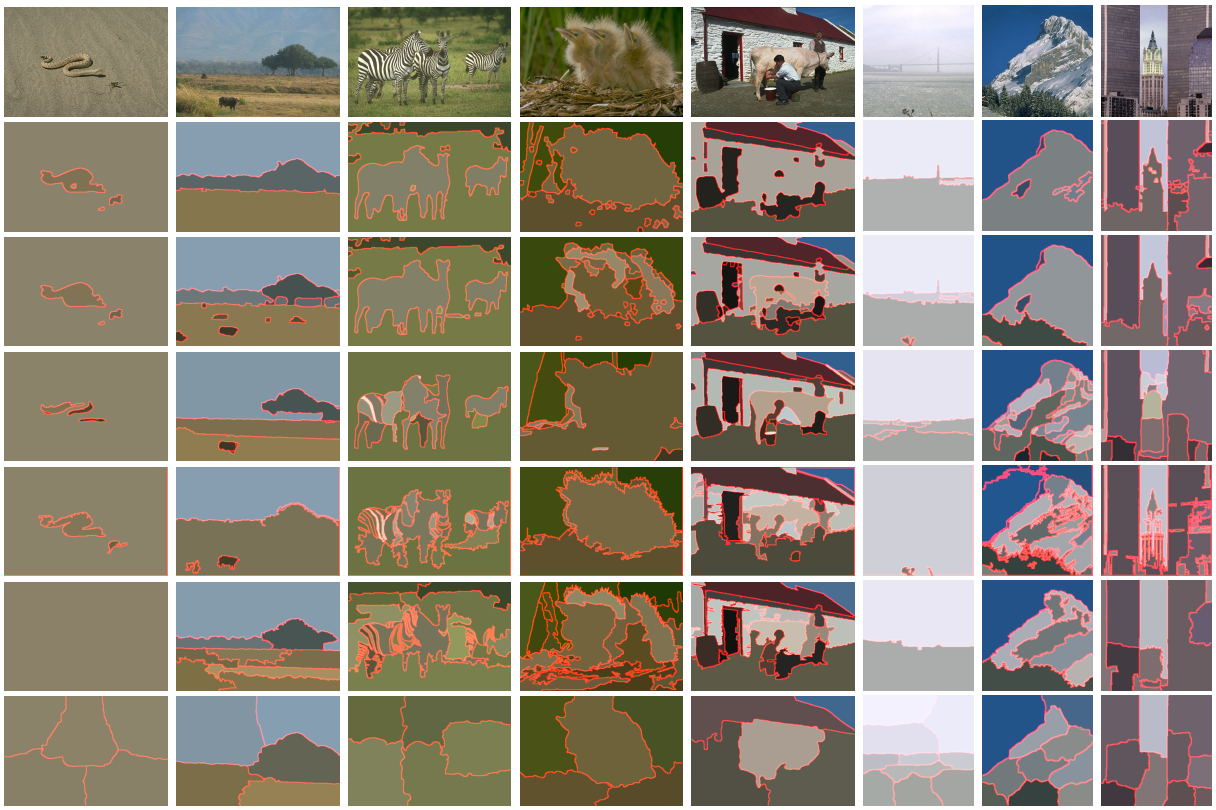

Figure 4. **Comparisons across models.** From Top to Bottom: PYdist, PYall, gPb, FH, MS, Ncuts

[5] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *PAMI*, 24(5):603–619, May 2002.

[6] D. Cremers, M. Rousson, and R. Deriche. A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape. *IJCV*, 72(2):195–215, 2007.

[7] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59(2):167–181, 2004.

[8] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *PAMI*, 6(6):721–741, 1984.

[9] D. Hoiem, A. A. Efros, and M. Hebert. Geometric context from a single image. In *ICCV*, volume 1, pages 654 – 661, 2005.

[10] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, pages 282–289, 2001.

[11] T. Malisiewicz and A. A. Efros. Improving spatial support for objects via multiple segmentations. In *BMVC*, September 2007.

[12] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, volume 2, pages 416–423, July 2001.

[13] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *PAMI*, 26:530–549, May 2004.

[14] T. P. Minka. Expectation propagation for approximate bayesian inference. In *UAI*, pages 362–369, 2001.

[15] R. Nock and F. Nielsen. Statistical region merging. *PAMI*, 26:1452–1458, November 2004.

[16] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001.

[17] J. Pitman and M. Yor. The two-parameter Poisson–Dirichlet distribution derived from a stable subordinator. *Annals of Probability*, 25(2):855–900, 1997.

[18] W. M. Rand. Objective criteria for the evaluation of clustering methods. *JASA*, 66:846–850, 1971.

[19] X. Ren and J. Malik. A probabilistic multi-scale model for contour completion based on image statistics. In *ECCV*, volume 1, pages 312–327, 2002.

[20] X. Ren and J. Malik. Learning a classification model for segmentation. In *ICCV*, volume 1, pages 10–17, 2003.

[21] B. C. Russell, A. A. Efros, J. Sivic, W. T. Freeman, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *CVPR*, pages 1605–1614, 2006.

[22] G. Sfikas, C. Nikou, and N. Galatsanos. Edge preserving spatially varying mixtures for image segmentation. *CVPR*, 0:1–7, 2008.

[23] J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22(8), 2000.

[24] A. Shyr, T. Darrell, M. I. Jordan, and R. Urtasun. Supervised hierarchical Pitman-Yor process for natural scene segmentation. In *CVPR*, pages 2281–2288, 2011.

[25] E. B. Sudderth and M. I. Jordan. Shared segmentation of natural scenes using dependent Pitman-Yor processes. In *NIPS*, pages 1585–1592, 2008.

[26] D. Sun, E. B. Sudderth, and M. Black. Layered image motion with explicit occlusions, temporal consistency, and depth ordering. In *NIPS*, pages 2226–2234. 2010.

[27] Z. Tu and S. Zhu. Image segmentation by data-driven Markov Chain Monte Carlo. *PAMI*, 24:657–673, 2002.

[28] J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *IEEE Tran. IP*, 3(5):625–638, Sept. 1994.

[29] M. Welling and K. Kurihara. Bayesian K-means as a "Maximization-Expectation" algorithm. In *SDM*, 2006.