Approximate Bayesian Computation for Distance-Dependent Learning

Soumya Ghosh
Disney Research, Boston
222 Third Street
Cambridge, MA 02142, USA
soumya.ghosh@disneyresearch.com

Erik B. Sudderth
Department of Computer Science
Brown University
Providence, RI 02912-1910, USA
sudderth@cs.brown.edu

Abstract

The distance dependent Chinese restaurant process (ddCRP) and its hierarchical extensions provide a flexible framework for clustering data with temporal, spatial, or other non-exchangeable dependencies. The successful application of these models crucially depends on functions chosen to encode structural dependencies exhibited by the data. Designing such affinity functions is challenging and often involves significant trial and error. Here, we explore methods for learning these functions from human annotated data. Leveraging recent advances in approximate Bayesian computation (ABC) we design algorithms that are effective at learning affinity functions from collections of human annotated image and video partitions and at achieving competitive results on standard benchmarks.

1 INTRODUCTION

Visual data exhibits strong spatial and temporal dependencies, and motivates methods capable of handling such correlations. Distance dependent models [1, 2, 3] leverage user specified affinities to model correlations in the data and provide flexible distributions over part based representations of images and videos. They represent partitions (or binary feature matrices) via links between data instances: each observation links to one other, and the probability of linking to nearby instances is higher. Closeness is measured according to affinities which may be arbitrarily specified to capture domain knowledge. The connected components of the induced link graph then partitions the dataset. When applied to visual data they produce representations useful for higher level tasks, for example, scene understanding.

Successful application of distance dependent models hinge critically on the specification of affinity functions. However, designing affinity functions can be challenging, especially for hierarchical variants [2] where affinities must be specified both among data instances and between latent clusters. As a result previous works [1, 4, 5, 6] have resorted to simple affinity functions that may be suboptimal at capturing complex dependencies exhibited by real world data. Here, we introduce feature augmented models that express affinities as linear combinations of user specified (weak) "cues" encoding similarities between data instances and between clusters. The feature weights capture the relative importance of different cues and are learned from human provided partitions. Since, only labeled partitions are observed and not the underlying links, innovations are necessary for approximately marginalizing over the exponentially large set of latent links. Further, the complex noise process employed by humans while explaining visual data produces partitions that exhibit high variance(Figure 1) and is challenging to model. We develop algorithms that leverage recent advances in approximate Bayesian computation (ABC) [7] to marginalize over the latent links without requiring an explicit specification of the noise process. Experiments on standard image and video benchmarks demonstrate the effectiveness of the proposed methods with learned models achieving results competitive with the state-of-the-art.

Figure 1: Human interpretations of natural images (and other visual data) exhibit wide variability. Here we have two images from the Berkeley segmentation dataset and corresponding segmentations produced by different expert annotators.

2 HIERARCHICAL DISTANCE DEPENDENT PARTITIONS

The distance-dependent CRP [1] defines a distribution over partitions through links between data instances. Each data point i has an associated latent link variable c_i which links to another data instance j, or itself, according to the distribution $p\left(c_i = j \mid A, \alpha\right) \propto \begin{cases} A_{ij} & i \neq j, \\ \alpha & i = j, \end{cases}$ where *affinity* $A_{ij} \in \mathbb{R}_0^+$, the set of non-negative real numbers. For notational convenience we denote $c_i \sim$ ddCRP(α, A) to denote the distribution of links. The resulting link structure induces a partition, where two data instances are assigned to the same cluster if and only if one is reachable from the other by traversing the link edges.

The hierarchical ddCRP (hddCRP) [2] defines a distribution over partitions of grouped data. It applies the ddCRP formalism twice, once for clustering data within a group into local clusters, and again for coupling local clusters across groups. The resulting distribution over partitions places higher probability mass on partitions that group nearby data points into latent clusters, *and* couple similar local clusters into global components. Given a collection of G groups with N_g observations each, we associate a latent data link with instance i in group g, c_{gi} which is distributed according to a group specific ddCRP(α_g, A^g). The connected components of the links $c_g = \{c_{gi} \mid i = 1, \ldots, N_g\}$ then determine the local clustering for group g. Data links $\mathbf{c} = \{c_1, \ldots, c_G\}$ across all groups divide the dataset into a set of local clusters $T(\mathbf{c})$. We further associate each cluster $t \in T(\mathbf{c})$ with a latent cluster link k_t drawn from a global (transcending groups) ddCRP distribution $p(k_t = s \mid \alpha_0, A^0(\mathbf{c})) \propto \begin{cases} A^0_{ts}(\mathbf{c}) & t \neq s, \\ \alpha_0 & t = s, \end{cases}$ where α_0 is a global self-affinity parameter, and $A^0(\mathbf{c})$ is the set of pairwise affinities between local clusters in $T(\mathbf{c})$. The connected components of $\mathbf{k} = \{k_t \mid t \in T(\mathbf{c})\}$ then couple local clusters into global components that are shared across groups. Since the data links $\mathbf{c}$ are conditionally independent given $A^{1:G}$, and cluster links $\mathbf{k}$ are conditionally independent given $\mathbf{c}$ and $A^0(\mathbf{c})$, the distribution induced over partitions, factorizes as,

$$p(\mathbf{c}, \mathbf{k} \mid \vartheta) = \prod_{g=1}^{G} \prod_{i=1}^{N_g} p(c_{gi} \mid \alpha_g, A^g) \prod_{k_t \in \mathbf{k}} p(k_t \mid \mathbf{c}, \alpha_0, A^0(\mathbf{c})), \tag{1}$$

with $\vartheta = \{\alpha_{1:G}, \alpha_0, A^{1:G}, A^0\}$.

3 LEARNING DISTANCE DEPENDENT MODELS

In this section, we present the main contribution of this paper – feature augmented distance dependent models that use linearly parametrized affinity functions. The affinity between data instances i and j is modeled as $A_{ij} = f(w_c^T \theta_{ij}^c)$, where $\theta_{ij}^c \in \mathbb{R}^M$ are user specified features, $w_c \sim \mathcal{N}(0, \Psi_c)$ and f is a monotonic nonlinear function that maps its argument to $\mathbb{R}_0^+$. The affinity between latent clusters t and s is similarly modeled as $A^0_{ts} = f(w_k^T \theta_{ts}^k),\ w_k \sim \mathcal{N}(0, \Psi_k)$. The features θ encode weak cues, such as the spatial distance or the strength of intervening contours between pixels.

The weight parameters $w = [w_c^T, w_k^T]^T$ need to be learned from annotated training partitions $Y = \{y_1 \ldots y_D\}$. A partition d containing N_d data instances is labeled with a vector $y_d \in \mathbb{N}^{N_d \times 1}$ encoding the allocation of data instances to partition elements. For example, if the first element of the partition belongs to a component labeled l then $y_{d1} = l$. Learning is complicated by two facts. First, our partitions are defined indirectly through links between data instances (and clusters). The mapping from links to partitions is many-to-one with exponentially many link combinations generating the same partition. To cope with this intractability we develop algorithms that approximately

marginalize over the exponentially large set of links and explore the marginal distribution $p(w \mid Y)$. The corresponding joint distribution $p(w, Y) = p(w)p(Y \mid w)$ requires the specification of the likelihood model $p(Y \mid w)$. This brings us to the second challenge. Human interpretations of images and videos vary wildly (Figure 1), exhibiting both large intra and inter annotator variance and the generative process employed by humans to partition visual data is difficult to model. We deal with this challenging issue by resorting to likelihood free approximate Bayesian computation(ABC) [7] techniques. ABC algorithms assume that simulation of the likelihood model is tractable even though the model itself might be intractable. Inferences about latent variables are then made by matching *summary* statistics of the simulated and observed data. We build on the efficient MCMC based ABC algorithms [7] that sample from a target distribution whose support is restricted to some neighborhood around the observed data. The target distribution for the hddCRP is,

$$\begin{aligned} p(\mathbf{c}, \mathbf{k}, w, Y) &\propto p(w) \prod_{d=1}^{D} p(\mathbf{c}_d \mid w_c) p(\mathbf{k}_d \mid \mathbf{c}_d, w_k) \delta(\Lambda(\mathbf{c}_d, \mathbf{k}_d), y_d), \\ \delta(y_a, y_b) &= \begin{cases} 1 & \text{if } \Delta(y_a, y_b) < \epsilon, \\ 0 & \text{otherwise,} \end{cases} \end{aligned} \tag{2}$$

where $\Lambda(\mathbf{c}_d, \mathbf{k}_d)$ represents the partition induced by links $\mathbf{c}_d$ and $\mathbf{k}_d$. The distribution's support is restricted to those simulated partitions that are at most ϵ away from human produced partitions Y. The notion of closeness is captured via a loss function $\Delta(y_a, y_b) = 1 - \text{RI}(y_a, y_b)$ that is invariant to arbitrary labelings of the partition. Here, $\text{RI}(y_a, y_b) \in [0, 1]$ measures the Rand index [8] between partitions y_a and y_b, with 1 indicating perfect agreement. The target distribution for the ddCRP model is analogous. With a sufficiently small threshold ϵ, the MCMC-ABC sampler produces samples from the marginal posterior density $p(w \mid Y)$ which is sufficiently concentrated around realizations of w that favor human annotated partitions.

We initialize the sampler with human annotated partitions by setting $\{\Lambda(\mathbf{c}_d, \mathbf{k}_d) = y_d\}_{d=1}^{D}$, instead of following the standard procedure [7, Algo. 3] of initializing via a rejection sampler that samples the prior distribution till a sample with non-zero probability (i.e., a partition within the threshold) is encountered. In the high dimensional space of partitions such an initialization procedure is extremely inefficient, and would render the entire algorithm ineffective. Conditioned on $\Lambda(\mathbf{c}_d, \mathbf{k}_d)$, we use a random walk Metropolis Hastings step with and accept proposal $w^* \sim \mathcal{N}(w, \nu\mathbf{I})$ with probability $\propto \min(1, \rho)$. Here ν is a free parameter controlling the scale of the proposals. The acceptance ratio ρ is,

$$\rho = \frac{p(\mathbf{c}, \mathbf{k}, w^*, Y) q(w_c, w_k \mid w_c^*, w_k^*)}{p(\mathbf{c}, \mathbf{k}, w, Y) q(w_c^*, w_k^* \mid w_c, w_k)} = \frac{p(w_c^*) \prod_d p(\mathbf{c}_d \mid w_c^*) p(\mathbf{k}_d \mid \mathbf{c}_d, w_k^*)}{p(w_c) \prod_d p(\mathbf{c}_d \mid w_c) p(\mathbf{k}_d \mid \mathbf{c}_d, w_k)} . \tag{3}$$

Conditioned on w^*, we propose links $\mathbf{c}^*, \mathbf{k}^*$[1]. Our MCMC-ABC sampler repeatedly iterates between these two sampling blocks. After running the sampler for a sufficiently long period we approximate the posterior marginal $p(w \mid Y)$ with a Monte-Carlo estimate and summarize it using its mode,

$$\hat{w} \approx \underset{w \in \{w^{(1)}, \ldots, w^{(S)}\}}{\text{argmax}} \sum_{s'=1}^{S} p(\mathbf{c}^{(s')}, \mathbf{k}^{(s')}, Y \mid w) p(w) . \tag{4}$$

Learning the feature augmented ddCRP model proceeds analogously via a MCMC-ABC sampler designed to explore the target density, $p(\mathbf{c}, w_c, Y) \propto p(w_c) \prod_{d=1}^{D} p(\mathbf{c}_d \mid w_c) \delta(\Lambda(\mathbf{c}_d), y_d)$. Again, we use a random walk MH proposal $w_c^* \sim \mathcal{N}(w_c, \nu_c \mathbf{I})$ for proposing weights governing data links $\mathbf{c}$. Conditioned on the weights we sample $\mathbf{c}$ using a straightforward Gibbs step, $c_{di} \mid \mathbf{c}_{-di}, w_c, Y \sim p(c_{di} \mid w_c) \delta(\Lambda(\mathbf{c}_d), y_d)$.

4 DISTANCE DEPENDENT MIXTURES

Armed with ABC based learning, the distance dependent models provide expressive distributions over partitions. When coupled with a data generating mechanism they provide a flexible tool for

[1] See Appendix for details

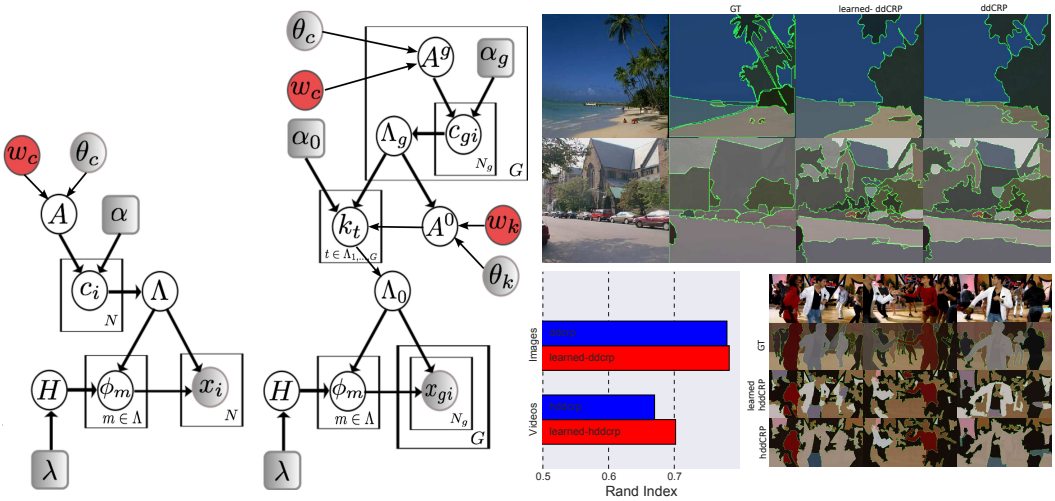

Figure 2: Graphical model representations for the feature augmented ddCRP and hddCRP models. In the ddCRP mixture, a partition is sampled from the feature augmented ddCRP prior with data links sampled according to $c_i \sim p(c_i \mid \alpha, A)$ and a connected components operation generating the partition Λ. Each component m of the partition is endowed with a parameter ϕ_m, sampled from a base distribution $H(\lambda)$, responsible for generating $x_i \sim \phi_m$ for $i \in m$. The affinities A are modeled via linear combination of features (θ), $A_{ij} = f(w_c^T \theta_{ij}^c)$, $w_c \sim \mathcal{N}(0, \Psi_c)$. The hddCRP model introduces a hierarchy, first sampling partitions Λ_g from group specific feature augmented ddCRPs and then sampling cluster links k_t, $t \in \Lambda_{1:G}$ from a cluster level feature augmented ddCRP, $k_t \sim p(k_t \mid \alpha_0, A^0(\mathbf{c}))$. Connected components of the cluster links define a partition of the dataset Λ_0. The nodes in red have been learned from human annotated partitions.

modeling partitions of visual data. Our generative procedure involves first sampling a partition. Conditioned on the partition structure, we endow each component m with data generating parameters $\phi_m \sim \mathrm{H}(\lambda)$ and generate an observation i in group g according to $x_{gi} \sim p(x_{gi} \mid \phi_{z_{gi}})$. The joint distribution of the model factorizes as,

$$p(\mathbf{x}, \mathbf{k}, \mathbf{c} \mid \vartheta, \hat{w}, \theta_c, \theta_k, \lambda) = p(\mathbf{c}, \mathbf{k} \mid \vartheta, \hat{w}, \theta_c, \theta_k) \prod_{m=1}^{M(\mathbf{c},\mathbf{k})} \int \prod_{gi \mid z_{gi}=m} p(x_{gi} \mid \phi_m)\, d\mathrm{H}(\phi_m \mid \lambda) \quad (5)$$

where z_{gi} encodes the global component membership of data instance x_{gi}, $M(\mathbf{c}, \mathbf{k})$ is the number of components induced by the cluster and data links, $\hat{w} = \{\hat{w}_c, \hat{w}_k\}$ indicate weights learned from human partitions and λ parametrizes the base distribution H. The ddCRP mixture models are similar but are defined for a single group and do not contain cluster links. Figure 2 provides the corresponding graphical models. MCMC algorithms presented in [1, 2] can be used to infer the posterior distributions over partitions.

5 APPLICATIONS

We use the learned dddCRP mixture models to extract segments from images and the hierarchical ddCRP to extract spatio-temporal regions from videos (with each video frame corresponding to a group). We compare them against distance dependent models that use hand crafted affinities and have previously been shown to achieve competitive segmentation performance. We benchmark the models on images from the eight natural scene categories [9] dataset available as part of the LabelMe [10] collection and on videos from VSB100 [11] dataset. For the Labelme images, we select 150 images from each of the eight categories and use 50 randomly chosen ones for training category specific ddCRP models. For VSB100, we use the provided train (60) / test(40) split. We use Rand index [8] to measure discrepancy from held out human annotated test segmentations. Figure 2 presents results achieved on these datasets. While we observe a small quantitative improvement for the ddCRP mixtures, the benefits of learning are clearly evident for the hierarchical models, where learning leads to significantly improved video segmentation performance.

References

[1] D. M. Blei and P. I. Frazier. Distance dependent Chinese restaurant processes. *J. Mach. Learn. Res.*, 12:2461–2488, November 2011.

[2] Soumya Ghosh, Michalis Raptis, Leonid Sigal, and Erik B Sudderth. Nonparametric clustering with distance dependent hierarchies. In *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence*, pages 260–270. Association for Uncertainty in Artificial Intelligence (AUAI), 2014.

[3] Samuel J Gershman, Peter Frazier, David M Blei, et al. Distance dependent infinite latent feature models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 37(2):334–345, 2015.

[4] Ivan Titov and Alexandre Klementiev. A bayesian approach to unsupervised semantic role induction. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 12–22. Association for Computational Linguistics, 2012.

[5] R. Socher, A. Maas, and C. D. Manning. Spectral Chinese restaurant processes: Nonparametric clustering based on similarities. In *AISTATS*, 2011.

[6] S. Ghosh, E. B. Sudderth, M. Loper, and M. J. Black. From deformations to parts: Motion-based segmentation of 3D objects. In *NIPS*, pages 2006–2014, 2012.

[7] Jean-Michel Marin, Pierre Pudlo, Christian P Robert, and Robin J Ryder. Approximate bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180, 2012.

[8] W. M. Rand. Objective criteria for the evaluation of clustering methods. *JASA*, 66(336):846–850, 1971.

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

[10] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: A database web-based tool for image annotation. *IJCV*, 77:157–173, 2008.

[11] Fabio Galasso, Naveen Shankar Nagaraja, Tatiana Jimenez Cardenas, Thomas Brox, and Bernt Schiele. A unified video segmentation benchmark: Annotation, metrics and analysis. In *IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[12] X. Ren and J. Malik. Learning a classification model for segmentation. *ICCV*, 2003.

[13] G. Mori. Guiding model search using segmentation. *ICCV*, 2005.

[14] C. Fowlkes, D. Martin, and J. Malik. Learning affinity functions for image segmentation: Combining patch-based and gradient-based approaches. *CVPR*, 2:54–61, 2003.

[15] D. R. Martin, C.C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Trans. PAMI*, 26(5):530–549, 2004.

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

[17] D. R. Martin, C.C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Trans. PAMI*, 26(5):530–549, 2004.

[18] J. Chang, D. Wei, and J. W. Fisher III. A Video Representation Using Temporal Superpixels. In *CVPR*, 2013.

A Appendix

A.0.1 Link Proposals

To simplify the exposition, we focus on a particular group g and denote c_{gi} as c_i. We also consider a single partition here and drop the explicit dependence on d from our notation. Let the current state of the sampler be $\mathbf{k}(\mathbf{c})$ and $\mathbf{c} = \{\mathbf{c}_{-i}, c_i = j\}$, so that i and j are members of the same cluster t_{ij}. Let $\mathcal{K}_{t_{ij}} = \{k_s \mid k_s = t_{ij}, s \neq t_{ij}\}$ denote the set of other clusters linking to t_{ij}. We assume that $\Lambda(\mathbf{c}, \mathbf{k})$ is within ϵ of a training partition y, with closeness measured according to Rand index.

Split? To construct our link proposal, we first set $c_i = i$. This may split current cluster t_{ij} into two new clusters, in which case we let t_i denote the cluster containing data i, and t_j the cluster containing formerly linked data j. Or, the partition structure may be unchanged so that $t_i = t_{ij}$.

Incoming links $k_s \in \mathcal{K}_{t_{ij}}$ to a split cluster are independently assigned to the new clusters with equal probability:

$$q_{\text{in}}(\mathcal{K}_{t_{ij}}) = \prod_{k_s \in \mathcal{K}_{t_{ij}}} \left(\frac{1}{2}\right)^{\delta(k_s, t_i)} \left(\frac{1}{2}\right)^{\delta(k_s, t_j)} = \left(\frac{1}{2}\right)^{|\mathcal{K}_{t_{ij}}|}. \tag{6}$$

The current outgoing link is retained by one of the split clusters, $k_{t_j} = k_{t_{ij}}$.

Propose Link Next, we propose an instantiation of c_i from the ddCRP prior distribution $q(c_i) = p(c_i \mid \alpha, A)$.

Merge? Let $c_i = j^*$ denote the proposed data link. Relative to the reference configuration in which $c_i = i$, this link may either leave the partition structure unchanged, or cause clusters t_i and t_{j^*} to merge into t_{ij^*}. In case of a merge, the new cluster retains the current outgoing link $k_{t_{ij^*}} = k_{t_{j^*}}$, and inherits the incoming links $\mathcal{K}_{t_{ij^*}} = \mathcal{K}_{t_i} \cup \mathcal{K}_{t_{j^*}}$.

If a merge does not occur, but t_{ij} was previously split into t_i and t_j, the outgoing link $k_{t_j} = k_{t_{ij}}$ is kept fixed. For newly created cluster t_i, we then propose a corresponding cluster link k_{t_i} from the prior over cluster links:

$$q_{\text{out}}(k_{t_i}) = p(k_{t_i} \mid \alpha_0, A^0(\mathbf{c}), \mathbf{c}). \tag{7}$$

Note that the proposal $c_i = j^*$ may leave the original partition unchanged if $c_i = i$ does not cause t_{ij} to split, and $c_i = j^*$ does not result in a merge. In this case, the corresponding cluster links are also left unchanged.

Combining the two pairs of cases above and restricting the resulting partition to be within some ϵ of a human annotated partition y gives us the proposal distributions,

$$q(\mathbf{c}^*, \mathbf{k}^* | \mathbf{c}, \mathbf{k}) = \begin{cases} q(c_i^*) q_{\text{in}}(\mathcal{K}^*_{t_{ij}}) & \text{split, merge,} \\ q(c_i^*) & \text{no split, merge,} \\ q(c_i^*) q_{\text{out}}(k^*_{t_i}) q_{\text{in}}(\mathcal{K}^*_{t_{ij}}) & \text{split, no merge,} \\ q(c_i^*) & \text{otherwise.} \end{cases} \tag{8}$$

If $\Delta(\Lambda(\mathbf{c}^*, \mathbf{k}^*), y) < \epsilon$, the proposed links $\mathbf{c}^*$ and $\mathbf{k}^*$ are accepted according to the Metropolis rule, wlse they are rejected.

A.1 Learning ddCRP weights

We first consider the covariate dependent ddCRP model. Here we have the following target distribution,

$$p(\mathbf{c}, w_c, Y) \propto p(w_c) \prod_{d=1}^{D} p(\mathbf{c}_d \mid w_c) \delta(z(\mathbf{c}_d), y_d), \tag{9}$$

We explore the posterior $p(\mathbf{c}, w_c \mid Y)$ by embedding a random walk Metropolis Hastings step within the ddCRP Gibbs sampler. We proceed by proposing w_c from a Gaussian distribution:

$$w_c^* \sim \mathcal{N}(w_c, \nu \mathbf{I}), \tag{10}$$

where ν is a free parameter controlling the scale of the proposals. The proposed w_c^* is accepted with probability $\propto \min(1, \rho_c)$ where ρ_c is:

$$\rho_c = \frac{p(\mathbf{c}, w_c^*, Y) q(w_c \mid w_c^*)}{p(\mathbf{c}, w_c, Y) q(w_c^* \mid w_c)} = \frac{p(w_c^*) \prod_d p(\mathbf{c}_d \mid w_c^*)}{p(w_c) \prod_d p(\mathbf{c}_d \mid w_c)}. \tag{11}$$

Next, we sample cluster links $\mathbf{c}$ using a Gibbs step:

$$\begin{aligned} c_{di} \mid \mathbf{c}_{-di}, w_c, Y &\sim p(c_{di} \mid \mathbf{c}_{-di}, w_c, Y) \\ &\sim p(c_{di} \mid w_c)\delta(z(\mathbf{c}_d), y_d). \end{aligned} \tag{12}$$

Neither sampling step involves evaluating the likelihood's normalization constant. After running the sampler for a sufficiently long period of time and collecting S samples, we can estimate the MAP sample $\hat{w}$,

$$\hat{w} \approx \underset{w \in \{w^{(1)},\ldots,w^{(S)}\}}{\text{argmax}} \sum_{s'=1}^{S} p(\mathbf{c}_d^{(s')}, Y \mid w)p(w) \; . \tag{13}$$

A.2 Image Segmentation Details

We model images as observed collections of "superpixels" [12], which are small blocks of spatially adjacent pixels. Given a collection of superpixels our aim is to find segments made up of superpixels homogeneous in appearance *and* whose size statistics loosely match with human annotated segments. Further, we restrict ourselves to the problem of single image segmentation with $G = 1$ and drop the explicit dependence on g from our notation. As a preprocessing step, we divide each image from the two datasets into approximately 1000 superpixels [12, 13] [2] using the normalized cut algorithm [14].[3]

A.2.1 Prior

We consider a few different ddCRP priors. First, for the fixed affinity version (*ddCRP*) we manually specify data affinities that encourage spatial neighbors not separated by strong intervening contours to connect to one another by setting $A_{ij} = (1 - b_{ij}) \times \mathbf{1}[i, j]$. Here, $0 \leq b_{ij} \leq 1$ is the maximum Pb [15] response along a straight line segment connecting the centers of superpixels i, j, and $\mathbf{1}[i, j]$ takes a value of 1 if i and j are spatial neighbors, and 0 otherwise. The self connection parameter α is set to 10^{-8}. Next, in order to learn the affinities we use signed distances between superpixel locations along x and y axes ($\delta x = r_i - r_j$, $\delta y = y_i - y_j$) as features encoding superpixel affinities. Here, r_i and y_i represent the x and y location of superpixel i. The relative importance of these structural features are learned from data. Together with b_{ij} they specify the *learned-ddCRP* prior over image partitions.

$$\begin{aligned} A_{ij} &= f(w, i, j) = (1 + exp(d_{ij}))^{-1} \times \mathbf{1}[i, j], \\ d_{ij} &= w_c^T \theta_{ij}^c = w_c^T [\frac{r_i - r_j}{R}, \frac{y_i - y_j}{Y}, b_{ij}]^T, \end{aligned} \tag{14}$$

where $R = \max(|r_i - r_j|)$ and $Y = \max(|y_i - y_j|)$.

Both ddCRP and learned-ddCRP, through their dependence on image contours, describe conditional priors on image partitions. We also consider a generative version that only considers superpixel locations: $\theta_{ij}^c = w_c^T [\frac{r_i - r_j}{R}, \frac{y_i - y_j}{Y}]^T$.

Qualitative comparisons Figure 3 illustrate partitions sampled from the learned ddCRP. We consider both generative and conditional affinities. The generative affinities learn more general characteristics of the scene category, for instance the tall buildings category contains partitions with vertical structures while the mountain category consists of more triangular structures. Conditional samples adapt to particular images and more closely reflect the particular structure of the image being conditioned on.

Figure 3 presents summary statistics computed from $10,000$ partitions sampled from learned generative affinities. We find that the Forest, Street and Inside city categories on average have a larger number of segments per partition. The ground truth partitions of these categories contain a large number of small segments, as a result we learn weights that prefer smaller segments. In contrast, the Coast and Highway category human partitions contain fewer but larger segments. This is again reflected in the learned weights, partitions of these categories contain fewer segments. We also find that the segment sizes in the learned partitions roughly follow a power law distribution, across all categories. This is a well known property exhibited by natural image segmentations [16].

A.2.2 Likelihood

We describe the texture of each superpixel via a local texton histogram [17], using band-pass filter responses quantized to 128 bins. A 120-bin HSV color histogram is used to describe the color of the superpixel. Each superpixel i is summarized via these histograms $x_i = \{x_i^c, x_i^t\}$. These histograms are treated as conditionally

[2] *www.cs.sfu.ca/~mori/*

[3] *www.eecs.berkeley.edu/Research/Projects/CS/vision/*

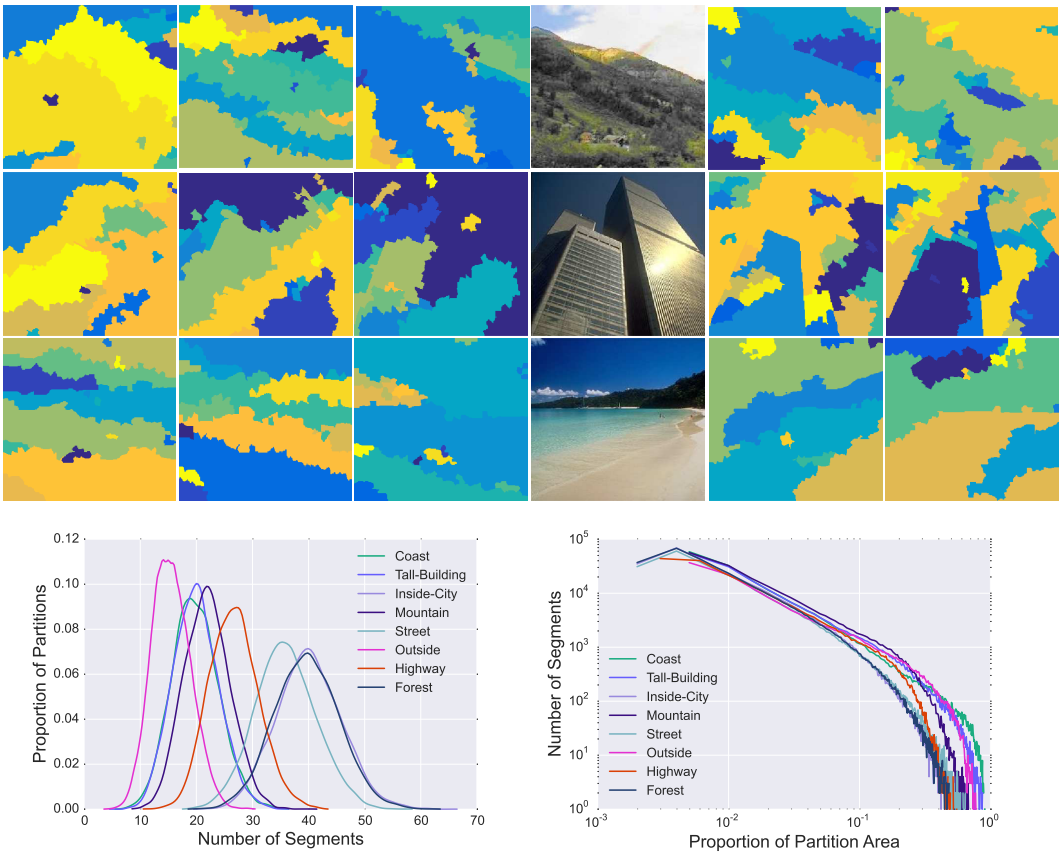

Figure 3: Samples from ddCRP **priors** with learned affinities. Rows display samples from a ddCRP model trained on the Mountain, Tall building, Coast categories. The first three columns correspond to generative samples while the two rightmost columns were generated by conditioning on the displayed image. *Bottom:* Summary statistics of partitions sampled from ddCRP models with learned generative weights. *Left:* Empirical distribution of the number of segments, broken down by the eight natural image category. *Right:* Number of segments occupying varying proportions of the image area, on a log-log scale.

independent given the cluster allocations z and are modeled as samples from multinomial distributions with Dirichlet priors.

$$x_i^c \sim \text{Mult}(\phi_{z_i}^c), \phi_{z_i}^c \sim \text{Dir}(\lambda^c), \qquad x_i^t \sim \text{Mult}(\phi_{z_i}^t), \phi_{z_i}^t \sim \text{Dir}(\lambda^t). \tag{15}$$

Hyperparameters The multinomial likelihoods treat pixels within a super-pixel as independent random variables. However, the ddCRP prior models affinities between superpixels. This can cause the prior to get washed away in favor of the likelihoods. To rectify this we introduce a hyperparameter γ that controls the relative importance of the prior and the likelihood,

$$p(\mathbf{x}, \mathbf{c} \mid \alpha, A, \gamma, \lambda) \propto p(\mathbf{c} \mid \alpha, A) \ \ \{p(\mathbf{x} \mid \mathbf{c}, \lambda)\}^\gamma. \tag{16}$$

The Dirichlet hyperparmaeters $\lambda = \{\lambda_c, \lambda_t\}$ along with γ are learned via a grid search on the training set. Given a grid of possible hyperparameters we hill climb on the posterior probability surface by running a small number of MCMC iterations. Finally, we select the set of hyperparameters that produce optimal results according to a chosen loss function, Rand index in this case. For the Dirichlet hyper-parameters we searched over a coarse grid located at locations: $\{0.01, 0.1, 1, 5, 10, 20, 25, 40, 50, 100\}$, for γ we searched over the range: $\{0.001, 0.005, 0.05, 0.01, 0.1, 1, 10\}$.

A.3 Video Segmentation Details

We consider the problem of discovering segments from videos that are coherent in space, time and appearance. The problem is a natural fit for the hierarchical ddCRP. We model video frames using independent spatial ddCRPs and couple them using a temporal ddCRP. As with image segmentation, instead of working with pixels we preprocess the video into a collection of superpixels.

A.3.1 Prior

The hddCRP prior requires affinity functions to be specified between both data instances and clusters. We experiment with both learned and manually specified affinity functions. In the learned case (*learned-hddcrp*), we reuse the image segmentation affinity functions between data instances. Affinity between clusters t, s is expressed as a linear weighted combination of covariates (θ^k_{ts}) encoding shape, size and positional affinities,

$$\theta^k_{ts} = [\vartheta_{ts}, \varphi_{ts}, \frac{|\zeta_t - \zeta_s|}{S}]^T. \tag{17}$$

The variable ζ_t denotes the size of cluster t and $S = \max|\zeta_t - \zeta_s|$. The covariates collectively represented by ϑ_{ts} capture within frame affinities and are defined as follows:

$$\vartheta_{ts} = \mathbf{1}_{[t,s|t\in g, s\in g]}[\frac{r_t - r_s}{R}, \frac{y_t - y_s}{Y}]^T. \tag{18}$$

Across frame affinities are captured in φ_{ts},

$$\varphi_{ts} = \mathbf{1}_{[t,s|t\in g+1, s\in g]}[\frac{|r_t - r_s|}{R}, \frac{|y_t - y_s|}{Y}, 1 - \frac{t \cap s}{t \cup s}]^T. \tag{19}$$

Within a frame we capture similarity between cluster locations using signed Manhattan distances. Across frame positional similarities are captured using standard Manhattan distances and through an intersection over union measure of the projection of one cluster on another. Finally, the affinity between clusters t, s is modeled via a sigmoidal transformation:

$$d_{ts} = w_k^T \theta^k_{ts}, \quad A^0_{ts} = (1 + exp(d_{ts}))^{-1}. \tag{20}$$

A.3.2 Likelihood

As a preprocessing step, we divide each frame into approximately 1200 superpixels using the method proposed by [18].[4] Following the image segmentation likelihood model, we describe a superpixel using 120-bin HSV color and 128-bin local texton histograms. The color and texture features for super-pixel i in video frame g are denoted by $x_{gi} = \{x^c_{gi}, x^t_{gi}\}$, where $x^c_{gi} \sim \text{Mult}(\phi^c_{z_{gi}}), \phi^c_{z_{gi}} \sim \text{Dir}(\lambda^c)$, $x^t_{gi} \sim \text{Mult}(\phi^t_{z_{gi}}), \phi^t_{z_{gi}} \sim \text{Dir}(\lambda^t)$. The proposed likelihood model forces clusters across video frames belonging to the same video segment share a common appearance model, encoding the assumption that appearance of objets doesn't change significantly over the course of the video. More elaborate likelihoods could be developed to capture appearance changes and is interesting future work. As with image segmentation, in addition to the Dirichlet hyperparameters controlling the texture and color likelihoods we introduce an additional parameter controlling the relative importance of the likelihood, $p(\mathbf{x}, \mathbf{k}, \mathbf{c} \mid \alpha_{1:G}, \alpha_0, A^{1:G}, A^0, \lambda) \propto p(\mathbf{c}, \mathbf{k} \mid \alpha_{1:G}, \alpha_0, A^{1:G}, A^0) \{p(\mathbf{x} \mid \mathbf{c}, \mathbf{k}, \lambda)\}^\gamma$. All likelihood hyperparameters are learned through validation analogously to image segmentation.

A.3.3 Additional Results

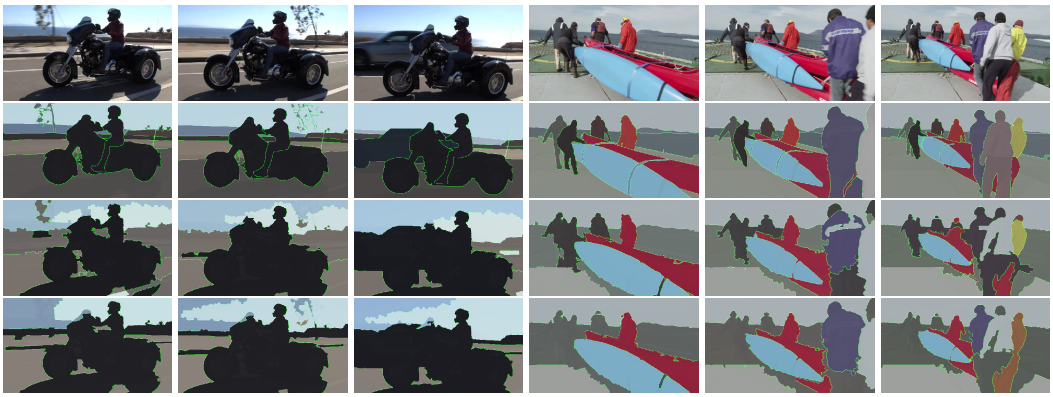

Figure 4: Examples from VSB100 test set. For each video the first, middle and last frames are displayed. The row immediately below the video displays the ground truth. The following two rows display segmentations produced by learned and naive-hddCRP models.

[4] [18] also estimate temporal correspondences between superpixels, but we do not utilize this information.