Temporally-Dependent Dirichlet Process Mixtures for Egocentric Video Segmentation

Joseph W. Barker, James W. Davis
Dept. of Computer Science and Engineering
Ohio State University, Columbus, OH 43210
barker.348@osu.edu, jwdavis@cse.ohio-state.edu

Abstract

In this paper, we present a novel approach for segmenting video into large regions of generally similar activity. Based on the Dirichlet Process Multinomial Mixture model, we introduce temporal dependency into the inference algorithm, allowing our method to automatically create long segments with high saliency while ignoring small, inconsequential interruptions. We evaluate our algorithm and other topic models with both synthetic datasets and real-world video. Additionally, applicability to image segmentation is shown. Results show that our method outperforms related methods with respect to accuracy and noise removal.

1. Introduction

The constantly decreasing cost of high-quality video cameras has inspired their inclusion in many professional environments, particularly for surveillance. One common need/task with stored long-duration video is to review (or summarize) the *predominant* activities in the video. Consider video cameras in police patrol cars and, more recently, on the officers themselves. Three (of many) activities that could be expected from the viewpoint of an officer are: 1) passing a citizen on the sidewalk, 2) giving a citizen directions, and 3) giving a citizen a ticket. If egocentric images of each of these activities were examined, the same visual feature will likely be seen: a person near the center of the frame. The differentiating factor is the temporal extent. Passing someone is only likely to occur for a few frames. Giving directions is likely to take longer. Finally, writing a ticket may often take minutes. In this case, given lower-level visual features, the *temporal extent* of the feature is key to determining the different classifications. Futhermore, even the importance may be temporally dependent, in that a passing citizen is largely irrelevant while writing a ticket is an important activity to log.

In this paper, we present a novel approach for taking such video and segmenting it into large temporal regions of generally similar activity. Our approach is to incorporate temporal dependence into the inference algorithm itself, and thus is capable of altering classification based on both temporal extent and the neighborhood around frames. This results in activity segments that are appropriately large and more contiguous.

We use the Dirichlet Process Multinomial Mixture (DPMM) model as the basis for our work. We modify it to include temporal (or, in general, distance) dependencies within the inference calculations. We evaluate and compare performance against competing models with temporal dependece as well as classic topic models.

2. Related Work

Originating in the area of document modeling, topic models are becoming popular in computer vision inference in part due to their strong grounding in Bayesian statistics and construction using histograms of features.

Similar to our goal and approach, Blei and Frasier [2] proposed the *distance-dependent* CRP (ddCRP), a modification to the well-known Chinese Restaurant Process. Thus, the temporal dependency is introduced at the clustering level, rather than at the data level, as in our approach. This makes it difficult for temporal similarity to overcome dissimilarity in data points. We will show in our comparative analysis that this causes their algorithm to have difficulty ignoring small noise regions.

Perhaps closest to our work is the 2-layer stacked DPMM algorithm proposed by Kitani *et al*. [12]. First, an online variant of DPMM is used to process the input data. The classifications are then divided into contiguous, non-overlapping, 12-element segments. Histograms constructed from each of these segments serve as the input to a second DPMM which outputs the final classification. Our method differs from this in several key ways. First, our windowing method is smooth and overlapping, allowing precise transitions between segments. Second, we allow the window size

to be varied (both manually and automatically). Finally, inference is performed simultaneously on both levels of our model, rather than in two sequential steps.

Other examples of such modifications to topic models include the dependent Dirichlet Process proposed by MacEachern [13] which, unfortunately, suffers the same inability to overcome data dissimilarity as previously discussed in ddCRP. Sticky HDP was proposed by Fox *et al.* [8] to address this, however they note that their method is only intended to prevent cycling between two simlar states. Our method is capable handling multiple, differing states.

Hospedales *et al.* [11] focus on short term activities which is diametrically opposed to our own objective (long segements). Specifically, they base their work on HMM's, which will be sensitive to short term variations. Emonet *et al.* [7] explicitly state that their work will be extremely sensitive to the temporal scale of the activities. We demonstrate the scale insensitivity of our algorithm in the experimental results.

Other examples of incorporating temporal relationships into inference algorithms include [1, 3, 4, 5, 6, 10, 17, 20].

3. Dirichlet Process Multinomial Mixtures

The Dirichlet Process Multinomial Mixture model is one member of the topic model family, that both learns the topic structure and uses the Dirichlet Process (DP) to induce a theoretically infinite but practically limited number of topics. Here we provide a basic overview of the compenents needed to derive our method. Readers interested in more details about the DP and related Gibbs samplers are directed to [18, 15].

3.1. The Dirichlet Process

The Dirichlet Process is essentially a distribution over distributions. It forms a countably infinite subset of a base distribution where each element of the subset is paired with a mixture probability. Alternately, sampling from the DP can also be specified as a *partition* over a discrete set:

$$\phi, z \sim \mathrm{DP}(\alpha_0 G_0) \tag{1}$$

where G_0 is the base distribution, α_0 is the concentration parameter (which controls the spread of probability across the subset and allows for bias), ϕ is a matrix (or vector of parameter sets) with each ϕ_i being a sample from the base distribution G_0 (i.e., ϕ is a subset of the domain of G_0), and z is the partition vector assigning each element of the set to one of the samples in ϕ. This is the *partition* view of DP, which results from the Chinese Restaurant Process.

3.2. Chinese Restaurant Process

The Chinese Restaurant Process (CRP) is a partitioning distribution, where a sample from it is a partition over a discrete set. It operates according to the following analogy: Assume a restaurant exists which contains an infinite number of tables. A (possibly) infinite series of customers enters the restaurant. Then, the probability of customer i sitting at table k is proportional to the number of customers already sitting at that table. Furthermore, the customer has a non-zero probability of sitting at an unoccupied table proportional to α_0. Finally, the CRP is exchangeable, meaning that any permutation of the customers (keeping the same assignments) will have equal probability. In other words, the order of arrival of the customers does not matter. Combining these, the final distribution is specified as:

$$P(z_i = k \mid z_{-i}) \propto \sum_{j \neq i} \mathbb{1}_{(z_j = k)} \tag{2}$$

$$P(z_i = k_{new} \mid z_{-i}) \propto \alpha_0 \tag{3}$$

where z_{-i} indicates all elements of z except i.

A model equivalent to the DP (partition view) using the CRP can be specified as:

$$\phi_k \sim G_0 \tag{4}$$

$$z \sim \mathrm{CRP}(\alpha_0) \tag{5}$$

where each element i of the set belongs to the topic with parameters ϕ_{z_i}.

3.3. DPMM

Given the above formulation, the DP mixture model using the multinomial distribution can be specified. First, each data element is assumed to have been sampled from a multinomial distribution:

$$x_i \mid z_i, \phi \sim \mathrm{Mult}(\phi_{z_i}) \tag{6}$$

Next, the parameter set ϕ needs a prior and we choose the common Dirichlet distribution (as it is the conjugate prior of the Multinomial distribution):

$$G_0 = \mathrm{Dirichlet}(\beta_0) \tag{7}$$

where β_0 is a parameter influencing the sparseness and/or structure of the topics.

Finally, the two are tied together using the DP:

$$\phi, z \sim \mathrm{DP}(\alpha_0 G_0) \tag{8}$$

This can then be rewritten using the CRP:

$$\phi_k \sim \mathrm{Dirichlet}(\beta_0) \tag{9}$$

$$z \sim \mathrm{CRP}(\alpha_0) \tag{10}$$

$$x_i \mid z_i, \phi \sim \mathrm{Mult}(\phi_{z_i}) \tag{11}$$

from which a Gibbs sampler can be derived. Of particular note is the "assignment" probability, the probability that a data point will be assigned to a specific topic k:

$$P(z_i = k \mid z_{-i}, \phi, x) \propto \left(\sum_{j \neq i} \mathbb{1}_{(z_j = k)} \right) \cdot \left(\prod_v (\phi_{kv})^{x_{iv}} \right) \tag{12}$$

The right-hand part of this equation comes from the multinomial distribution and is a measure of the similarity between a data element and a topic. The left-hand part comes from the CRP and is responsible for the clustering effect in the algorithm. Specifically, if the data element is similar to one (or more) of the topics, it will most likely be assigned to the topic among these with the most members. Conversely, a *new* topic may be created if it is sufficiently dissimilar to any topics to overcome the influence from the topic size. In general, we refer to these two parts as the data and clustering components of the algorithm.

4. Temporal-Dependence in DPMM

As previously discussed, temporal relationships are a key element in video segmentation. However, there is no temporal dependence in the standard DPMM (required for exchangeability). Hence, it becomes necessary to modify the model. There are two locations in the model where it is natural to consider adding such dependence, corresponding precisely to the clustering and data components.

4.1. Recursive Dirichlet Process Multinomial Mixtures (RDPM)

Temporal dependence can be incorporated into DPMM via the data component of the model, and is the approach we propose in this work. The weighting concept is similar to the ddCRP, but the question remains as to what should be weighted. A naïve choice might be to weight the data histograms themselves. That is, the weighting function would be applied as a filter over the input data. However, this approach effectively creates mixtures of topics, something the basic DPMM-based models have difficulty handling.

Instead, consider feeding the output of one DPMM into another DPMM in a recursive manner. The topic assignments from the lower layer are used to create a series of histograms, where each data element has a corresponding histogram constructed from its lower-level DPMM assignment *and those of its temporal neighbors*. This is similar in concept to the 2-layer *stacked* DPM-OL proposed by [12], but we will later show the important differences.

Making the modification yields the following model:

$$H_0 = \text{Dirichlet}(\lambda_0) \tag{13}$$

$$G_0 = \text{Dirichlet}(\beta_0) \tag{14}$$

$$w, \theta \sim \text{DP}(\gamma_0 H_0) \tag{15}$$

$$\phi_k \sim G_0 \tag{16}$$

$$z \mid w, \theta \sim \text{Steerable-CRP}(w, \theta, \alpha_0, \lambda_0, f, d) \tag{17}$$

$$x_i \mid z_i, \phi \sim \text{Mult}(\phi_{z_i}) \tag{18}$$

where w and θ are the partitioning and parameters for the second-level DPMM. The distribution referred to in Eqn. 17 as Steerable-CRP takes the population density-based clustering of the standard CRP and incorporates a bias that steers the clustering towards density templates θ_c chosen by the selection vector w. Further, the target density is a combination of all templates, weighted using distance d and windowing function f. The full distribution is as follows:

$$P(z_i = k \mid z_{-i}, w, \theta, \lambda_0) \propto \left(n_k - \mathbb{1}_{(z_i=k)}\right) \left[(C\lambda_0 + 1) \prod_c (\theta_{ck})^{\left(\sum_j f(d_{ij}) \mathbb{1}_{(w_j=c)}\right)\left(\sum_j f(d_{ij})\right)^{-1}}\right] \tag{19}$$

$$P(z_i = k_{new} \mid z_{-i}, w, \theta, \lambda_0) \propto \alpha_0 \lambda_0 \tag{20}$$

where C is the number of unique values in w.

The Gibbs sampler for this model can be derived as:

$$P(w_i = c \mid w_{-i}, \theta, H_0, z) \propto \left(m_c - \mathbb{1}_{(w_i=c)}\right) \cdot \prod_k (\theta_{ck})^{\left(\sum_j f(d_{ij}) \mathbb{1}_{(z_j=k)}\right)\left(\sum_j f(d_{ij})\right)^{-1}} \tag{21}$$

$$P(w_i = c_{new} \mid w_{-i}, \theta, H_0, z) \propto \gamma_0 \frac{\Gamma(K\lambda_0)}{\Gamma(\lambda_0)^K} \cdot \frac{\prod_k \Gamma\left(K\lambda_0 + \left(\sum_j f(d_{ij}) \mathbb{1}_{(z_j=k)}\right)\left(\sum_j f(d_{ij})\right)^{-1}\right)}{\Gamma(K\lambda_0 + 1)} \tag{22}$$

$$\theta_c \mid w, H_0, z \sim \text{Dirichlet}\left(\lambda_0 + \sum_i y_i \, \mathbb{1}_{(w_i=c)}\right) \tag{23}$$

$$P(z_i = k \mid z_{-i}, \phi, w, x_i) \propto \left[\left(n_k - \mathbb{1}_{(z_i=k)}\right) \prod_v (\phi_{kv})^{x_{iv}}\right] \left[(C\lambda_0 + 1) \prod_c (\theta_{ck})^{\left(\sum_j f(d_{ij}) \mathbb{1}_{(w_j=c)}\right)\left(\sum_j f(d_{ij})\right)^{-1}}\right] \tag{24}$$

$$P(z_i = k_{new} \mid z_{-i}, \phi, w, x_i) \propto \left[\alpha_0 \frac{\Gamma(V\beta_0)}{\Gamma(\beta_0)^V} \frac{\prod_v \Gamma(\beta_0 + x_{iv})}{\Gamma\left(V\beta_0 + \sum_v x_{iv}\right)}\right] [\lambda_0] \tag{25}$$

$$\phi_k \mid z, x \sim \text{Dirichlet}\left(\beta_0 + \sum_i x_i \, \mathbb{1}_{(z_i=k)}\right) \tag{26}$$

where

$$m_c = \sum_i \mathbb{1}_{(w_i=c)}, \quad n_k = \sum_i \mathbb{1}_{(z_i=k)}, \quad y_{ik} = \mathbb{1}_{(z_i=k)}, \quad K = \sum_k 1, \quad C = \sum_c 1 \tag{27}$$

We refer to our proposed method as Recursive DPMM, or RDPM.

This formulation has advantages over ddCRP, classic DPMM, and the 2-layer stacked DPM-OL. Our formulation can easily ignore noise (small, short-duration regions differing from the surrounding larger regions) as they only have a small effect on the second-level histogram. Where the 2-layer stacked DPM-OL approach uses a fixed-size, non-overlapping window, our approach uses overlapping windows. Also, as our approach explicitly links the two levels, the possibility exists for the second level to influence the first level. In other words, the second-level topic assignment can affect which first-level topic assignment is chosen. Finally, and importantly, inference on the parameters for the windowing function is possible, which we describe next.

4.2. Window Parameter Inference

The most common choices for a windowing function (Gaussian, Laplace, etc.) have at least one parameter. This unfortunately means that yet another parameter is needed for the model. However, we provide a method to do inference on the additional window parameter(s). As before, there are two approaches we might take to accomplish this, corresponding to the clustering and data components of the model.

Considering the data component, it would seem that the best window parameter value should be the one which maximizes the likelihood of each second-level histogram. However, in the case of a window *size* parameter (the most common type), this has the effect of driving the size to 0. This results from the fact that, for a given topic, the most likely histogram is the most pure histogram (concentrated in a single category).

Alternately, we can approach the problem from the perspective of the clustering component. It is desirable that the algorithm should a) use the smallest reasonable number of topics, and b) form the largest possible contiguous regions. We can achieve both goals if a parameter is more likely when it would cause an element to move into a topic with more members (within the contraints of the main model). This is accomplished by adding an element to the sampler:

$$\begin{aligned} P(\rho \mid w, \theta, z) &\propto P(\rho) P(w \mid z, \theta, \sigma) \\ &\propto P(\rho) \prod_i m_{\tilde{w}_i(\rho)} \end{aligned} \tag{28}$$

where

$$\tilde{w}_i(\rho) = \arg\max_c \left(m_c \prod_k (\theta_{w_i k})^{\sum_j f(d_{ij},\, \rho) \mathbb{1}_{(z_j = k)}} \right) \tag{29}$$

is the maximum-likelihood topic assignment when using parameter ρ.

Note that this does not give a closed form distribution and thus an approximate sampling technique such as importance sampling or Metropolis-Hastings will be required.

The end result is that, in the case of window size, the parameter will be increased as long as the size of small regions (relative to the parameter) can be reduced and will stop increasing when only large regions remain.

5. Experiments

In order to test the efficacy of the proposed temporally-dependent RDPM algorithm, a series of experiments using synthetic and real video sequences were explored. Results are compared to several common approaches. We additionally show the applicability of the algorithm to the task of image segmentation.

5.1. Synthetic Dataset 1: NOISE

This dataset was intended to test performance on handling noise-like events: small regions of one topic contained inside larger regions of another. The *partition* vector z (indicating the topic to which each data point belongs) was set to contain a sequence of equal-sized regions of different topics. In the center of each region, a small noise region of a differing topic was placed. The size of the small region increases as the sequence proceeds, which allowed the level of noise rejection to be measured. A visualization of the NOISE dataset and the corresponding ground truth labels (desired regions) is shown in Fig. 2 (left, top and bottom rows).

The actual data points are generated by sampling from the generative model:

$$x_i \sim \text{Mult}(\phi_{z_i}), \quad \phi_k \sim \text{Dirichlet}(0.1) \tag{30}$$

with the added restriction that for every pair of topics k and j, the KL-Divergence between their paramaters, ϕ_k and ϕ_j, was greater than 3.

5.2. Synthetic Dataset 2: PERIODIC

The second dataset, PERIODIC, was created using a similar method to the first, but modified to allow mixtures of topics. This is a challenging scenario for topic models and DPMM in particular. Ground truth remains a sequence of equal-sized regions. Odd numbered regions are also still generated from a single topic. However, even numbered regions are generated from a mixture of the topics of the two neighboring regions:

$$x_i \sim h(i)\,\text{Mult}(\phi_{z_i}) + (1 - h(i))\,\text{Mult}(\phi_{y_i}) \tag{31}$$

where the mixture function $h(\cdot)$ is a modified sine wave of slowly increasing frequency:

$$h(i) = \frac{1}{2} sin(c \cdot i + 1) + \frac{1}{2} \tag{32}$$

A visualization of the PERIODIC dataset and ground truth can be seen in Fig. 2 (right, top and bottom rows). Note that, for the purposes of visualization, only the mixture topic with highest proportion is shown.

(a) (b)

Figure 1. Egocentric video sequence. (a) Sample frames and (b) Sample depth map.

5.3. Egocentric Video Sequence

The next dataset comes from a 7 min egocentric stereo video sequence of a walk around a college campus, including both indoor and outdoor scenes (see Fig. 1.a). The video was recorded using a Sony HDR-TD10 3D camcorder. A depth map was constructed (using OpenCV) from the video which served as the basic input sequence for testing (see Fig. 1.b). The depth at each pixel was quantized into one of three regions: less than 4 feet (near), greater than 14 feet (far), or in-between (mid). Finally, a 3-bin histogram for each image was constructed over these depth categories. This is a simple feature set (compared to the motion features of [12]), but it was selected for two reasons. First, it is more challenging for an inference algorithm to achieve adequate performance using a simple feature set versus a sophisticated one, making performance of each algorithm more apparent. Second, initial testing showed that even with this simple feature set, it was possible to find video segments of consistent activities.

5.4. Image Segmentation

The final dataset was selected to demonstrate the applicability of the method to other tasks, such as image segmentation. This makes use of the fact that RDPM (as well as ddCRP and DPM-OL) are distance-based methods, and thus should work regardless of whether *temporal* distance or *spatial* distance is used. Similar to the work of Gosh [9], we select images from the LabelMe database [19]. Approximately 1000 superpixels are extracted [14] and summarized using 64-bin color histograms. Ground truth is also assigned to each superpixel using the labeling provided by LabelMe.

5.5. Algorithms

We initially tested the temporal datasets with our RDPM method, using a Gaussian window function with spread parameter $\sigma^2 = 20$ (unless otherwise specified). We then tested the variant of our method where σ^2 is inferred.

For comparison, 4 other algorithms were examined. The first is a straightforward K-means implementation. The K parameter was set to the known number of topics for the synthetic datasets. For the video sequence, K was set to 6 (the number of topics detected by the best performing method). The second comparison algorithm was an implementation of the DPMM model described in Section 3.3. The third algorithm was the 2-layer online DPMM (DPM-OL) model with clip size of 12 (or a 10x10 grid for image segmentation), as proposed by [12]. The final algorithm was a Gibbs sampler for the Hierarchical ddCRP Multinomoial mixture model (H-ddCRP) [2], using a Gaussian windowing function with parameter $\sigma^2 = 20$. The default value of $\sigma^2 = 20$ was selected for our approach and H-ddCRP to closely match the 2-layer DPM-OL window size of 12. For any parameters not explicitly mentioned above (e.g., α_0), several values were tested and the value giving the best result was kept.

NOISE	Accuracy	Noise Removal	Rand Index	Avg Seg Len
K-means	88.4%	0%	0.744	184
DPMM	88.4%	0%	0.744	184
2-layer DPM-OL	89.3%	57.5%	0.735	318
H-ddCRP	88.4%	0%	0.744	184
RDPM, fixed $\sigma^2 = 20$	93.2%	41.7%	0.843	267
RDPM, inferred σ^2	99.9%	100%	0.998	552
Ground truth	100%	100%	1	552

Table 1. Accuracy, noise removal rate, Rand index, and average segment length on dataset NOISE.

6. Results

6.1. NOISE Dataset

Figure 2 shows the final segmentations for dataset NOISE over the set of algorithms. As noise removal is the primary concern of this experiment, we show the fraction of total noise removed in Table 1. Additionally, we provide the accuracy and Rand index [16], a measure of clustering similarity related to accuracy. Finally, since our final goal is to extract long segments, we provide the average segment length.

We see that K-means, DPMM, and H-ddCRP perform essentially identically. They recover the base topic regions but also recover the small noise regions, meaning no noise removal.

The 2-layer DPM-OL algorithm demonstrates better per-

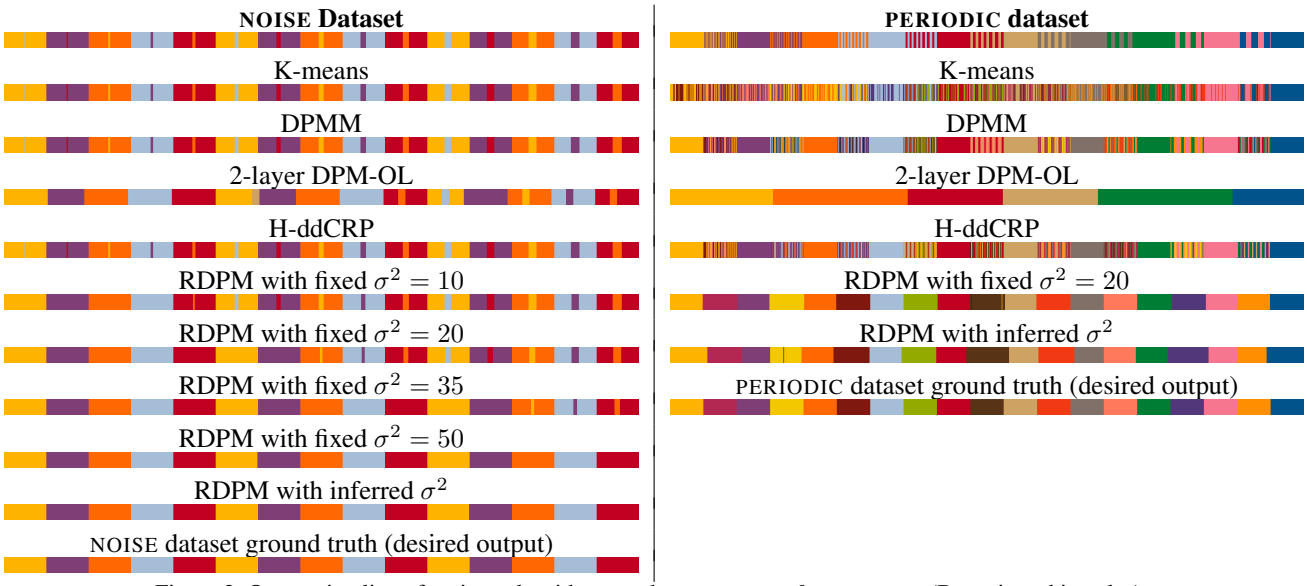

Figure 2. Output timeline of various algorithms on datasets NOISE & PERIODIC. (Best viewed in color)

formance, removing over half of the noise. However, there is only a slight increase in accuracy. On the borders of regions, issues caused by the non-overlapping windows results in several borders being shifted (aliasing) and one border being misclassified. This counteracts much of accuracy gained from noise removal. We also note that the noise removal is inconsistent, in that one larger noise region is removed while two smaller regions are kept. While increasing the clip size (above 12) would improve noise removal, it would also exacerbate the aliasing problem. Conversely, reducing the clip size would reduce the aliasing problem but also decrease noise removal.

With $\sigma^2 = 20$, the proposed RDPM achieves somewhat better accuracy but lower noise removal than the 2-layer DPM-OL. However, none of the border issues that plagued the 2-layer DPM-OL are present. As shown in Fig. 2, using values of σ^2 other than the default demonstrates that the amount of noise removal can be precisely controlled. Slowly increasing σ^2 causes a steady, consistent decrease of noise in the final result. Finally, at $\sigma^2 = 50$, all the noise is removed.

The RDPM variant with the inferred window parameter provides similar results to $\sigma^2 = 50$. With the inferred parameter ($\sigma^2 = 101.3$), the accuracy is near perfect and 100% of the noise is removed. One may note that the inferred parameter value is larger than stricly required. This is due to the fact that the inference mechanism requires the parameter to grow as long as large regions do not shrink. There is no requirement that the parameter must be as small as possible. In other words, the parameter will grow to the largest size that does not cause adverse effects. Finally, as expected, we see that average segment length increases with accurracy.

PERIODIC	Acc.	Rand Index	Avg Seg Len
K-means	21.7%	0.387	14.6
DPMM	67.2%	0.539	24.8
2-layer DPM-OL	31.6%	0.391	1267
H-ddCRP	65.8%	0.544	22.9
RDPM, fixed $\sigma^2 = 20$	98.1%	0.962	330
RDPM, inferred σ^2	92.6%	0.856	330
Ground truth	100%	1	400

Table 2. Accuracy, Rand index, and average segment length for various algorithms on dataset PERIODIC.

6.2. PERIODIC Dataset

With the PERIODIC dataset, we expect most algorithms to have difficulty adapting to the complex mixture of topics. Since this experiment is concerned with the classification of each region, we exclude the noise removal measure. Table 2 provides the results.

In Fig. 2, we see that the K-means algorithm has extreme difficulty with this dataset. Due to the similarity between the mixed regions and their neighbors, the algorithm has difficulty even classifying the simple single topic regions. This gives very poor results (see Table 2).

DPMM has little difficulty recovering the single topic regions, but is unable to handle the mixed regions. Rather than each mixed region becoming a single topic, multiple topics are created matching various mixture proportions. The end result is an accuracy of 67%. The 2-layer DPM-OL algorithm also has significant difficulty, but in the opposite manner. It overgeneralizes and collapses the 19 true regions into 6. This results in a very low accuracy of 32%. H-ddCRP again shows essentially the same results as DPMM, recovering the single topic regions but failing on the mixed

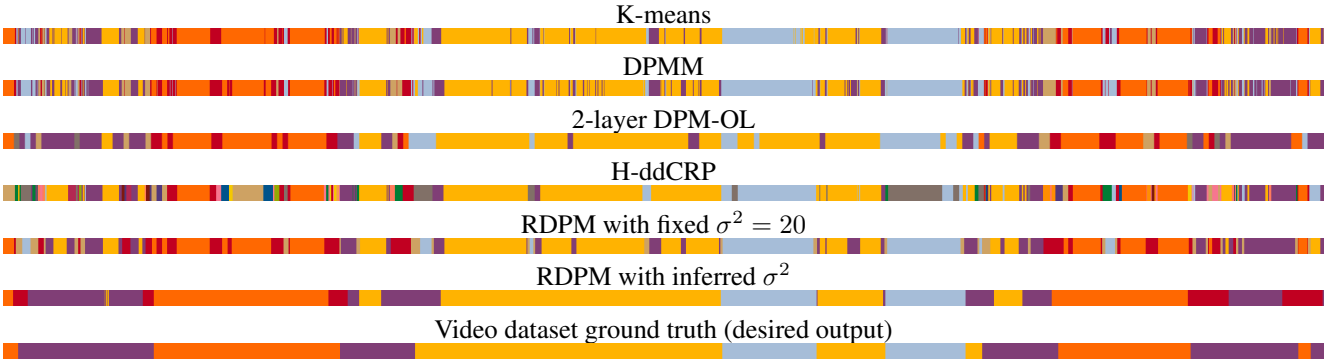

Figure 3. Output timeline of various algorithms on video sequence. (Best viewed in color)

Video Sequence	Rand Index	Avg Seg Len
K-means	0.393	36.8
DPMM	0.367	35.6
2-layer DPM-OL	0.428	153.7
H-ddCRP	0.353	63.7
RDPM, fixed $\sigma^2 = 20$	0.413	97.7
RDPM, inferred σ^2	0.674	397.5
Ground truth	1	823.4

Table 3. Rand index and average segment length for various algorithms on video dataset.

regions.

Compared to the previous approaches, our RDPM performs extremely well on this challenging dataset. With $\sigma^2 = 20$, it nearly perfectly classifies both single topic and mixed regions, with an accuracy of 98%. Interestingly, using the inferred σ^2 in this case results in slightly poorer (but still strong) performance. We attribute this to the algorithm being overly aggressive in selecting a larger window size ($\sigma^2 = 572.0$), and may indicate the need for a limiting factor in the parameter inference mechanism. This will be examined in more detail in a future work.

6.3. Real Video

As discussed previously, we desire long, contiguous regions which ignore/remove small, inconsequential interruptions. From visual inspection of the results in Fig. 3, we can see several obvious regions of long term activity (for example, the middle yellow region in the sequence) common across the algorithms. Comparison with the source video shows that these regions correspond with general activities such as walking indoors, walking outdoors, and following someone. The small regions interspersed within these larger regions are typically events such as a pedestrian crossing the path of the camera. In this work we consider such events inconsequential (though other tasks may desire such events), therefore our evaluation criteria is that these small regions should be eliminited while the large, semantically-coherent regions should be as long and uninterrupted as possible.

The K-means and DPMM algorithms are able to extract several large regions of similar activity. However, these regions are broken by many small, short-duration interruptions. The 2-layer DPM-OL, H-ddCRP, and RDPM (σ^2) algorithms are better able to smooth over the small interruptions in the large regions. However, in the smaller regions there are still a number of interruptions. With this dataset we see the clear benefit of using our RDPM method with parameter inference. Using the inferred window parameter ($\sigma^2 = 615.8$), only the largest regions remain giving the desired longer, continuous video segments.

To confirm these observations, we identified the major activities in the video sequence (such as indoor/outdoor travel and following behind an individual in a crowd) and compared the output of each algorithm to this ground truth. The results in Table 3 confirm our observations, specifically that our proposed algorithm produces longer segments (on average) and better clustering than competing methods.

6.4. Image Segmentation

In the context of image segmentation, our goal of long, uninterrupted sequences translates to large, contiguous regions. As shown in Fig. 4 (and Table 4), the results demonstrate that the three methods do a reasonable job of segmentation. Similar to the video experiment above, both our own RDPM and DPM-OL produce large segments. Again, as above, H-ddCRP shows more attention to the details in the scene and thus an inability to bridge over those details to create large regions. On the other hand, the windowing (grid) required by DPM-OL causes the segmentation to be blocky and thus results in a lower Rand index (segmentation similarity) value vs. H-ddCRP. Our proposed RDPM method produces smoother segmentations, with large average size, fewer interruptions and with a higher Rand index than competing methods.

The primary issue in the results produced by RDPM is that it may tend to oversmooth the region boundaries. For example, the corners of the red regions in the top image have been rounded off. This is due to a large windowing function and the fact that the algorithm will tend to seek

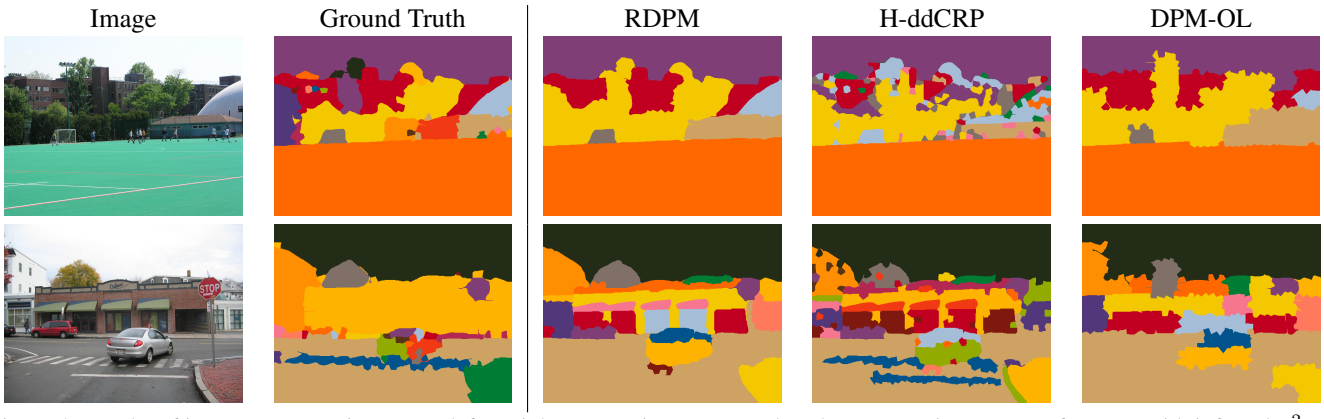

Figure 4. Results of image segmentation. From left to right: source image, ground truth segmentation, output of RDPM with inferred σ^2, output of H-ddCRP, and output of 2-layer DPM-OL. (Best viewed in color)

Image Segmentation	Rand Index		Avg Region Size	
	Img. 1	Img. 2	Img. 1	Img. 2
2-layer DPM-OL	0.779	0.672	289130	53333
H-ddCRP	0.838	0.696	56497	25263
RDPM, inferred σ^2	0.908	0.717	307200	45714

Table 4. Rand index and average region size for various algorithms on image dataset.

a 50/50 split between categories at border. Thus, the algorithm will adjust borders so that they are as smooth as possible.

Overall, the proposed RDPM approach has shown improved performance over K-means, DPMM, 2-layer DPM-OL and H-ddCRP on the various complex datasets evaluated, including video (temporal) and image (spatial) segmentation tasks.

7. Summary

In this paper we have proposed a modification to the DPMM topic model designed to introduce temporal dependence into the inference process. This is accomplished by layering two DPMMs, where the input of one is histograms (weighted by temporal distance) constructed from the ouput of the other. Results on multiple datasets (video and images) have shown that the proposed algorithm outperforms other state-of-the-art algorithms.

This research was supported in part by AFRL under contract No. FA8650-07-D-1220.

References

[1] A. Ahmed and E. P. Xing. *Dynamic non-parametric mixture models and the recurrent Chinese restaurant process*. CMU, School of Comp. Sci., 2007. 2

[2] D. Blei and P. Frazier. Distance dependent Chinese restaurant processes. *J. Mach. Learn. Res.*, 12:2461–2488, 2011. 1, 5

[3] F. Caron, M. Davy, and A. Doucet. Generalized Polya urn for time-varying Dirichlet process mixtures. *arXiv:1206.5254*, 2012. 2

[4] Y. Chung and D. B. Dunson. The local Dirichlet process. *Annals of the Institute of Statistical Mathematics*, 63(1):59–80, 2011. 2

[5] D. Dahl. Distance-based probability distribution for set partitions with applications to Bayesian nonparametrics. *JSM Proc., Sect. on Bayes. Stat. Sci., ASA*, 2008. 2

[6] J. Duan, M. Guindani, and A. Gelfand. Generalized spatial Dirichlet process models. *Biometrika*, 94(4), 2007. 2

[7] R. Emonet, J. Varadarajan, and J.-M. Odobez. Extracting and locating temporal motifs in video scenes using a hierarchical non parametric bayesian model. In *CVPR*, 2011. 2

[8] E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky. An HDP-HMM for systems with state persistence. In *ICML*, 2008. 2

[9] S. Ghosh, A. Ungureanu, E. Sudderth, and D. Blei. Spatial distance dependent Chinese restaurant processes for image segmentation. In *NIPS*, 2011. 5

[10] J. Griffin. Order-based dependent Dirichlet processes. *JASA*, 101(473):179–194, 2006. 2

[11] T. Hospedales, S. Gong, and T. Xiang. A markov clustering topic model for mining behaviour in vid. In *CVPR*, 2009. 2

[12] K. Kitani, T. Okabe, Y. Sato, and A. Sugimoto. Fast unsupervised ego-action learning for first-person sports videos. In *CVPR*, 2011. 1, 3, 5

[13] S. MacEachern. Dependent nonparametric processes. In *ASA Proc., Sect. on Bayes. Stat. Sci.* Alexandria, VA, 1999. 2

[14] G. Mori. Guiding model search using segmentation. In *ICCV*, 2005. 5

[15] R. Neal. Markov chain sampling methods for Dirichlet process mixture models. *J. of Comp. & Grph. Stat.*, 9(2):249–265, 2000. 2

[16] W. Rand. Objective criteria for the evaluation of clustering methods. *JASA*, 1971. 5

[17] L. Ren, D. B. Dunson, and L. Carin. The dynamic hierarchical Dirichlet process. In *ICML*, 2008. 2

[18] C. Robert. *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation*. Springer, 2007. 2

[19] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. *IJCV*, 2008. 5

[20] Y. Xue, D. Dunson, and L. Carin. The matrix stick-breaking process for flexible multi-task learning. In *ICML*, 2007. 2