NONPARAMETRIC IMAGE INTERPOLATION AND DICTIONARY LEARNING USING SPATIALLY-DEPENDENT DIRICHLET AND BETA PROCESS PRIORS

John Paisley, Mingyuan Zhou, Guillermo Sapiro[†] and Lawrence Carin

Department of Electrical & Computer Engineering
Duke University, Durham, NC, USA

[†]Department of Electrical & Computer Engineering
University of Minnesota, Minneapolis, MN, USA

ABSTRACT

We present a Bayesian model for image interpolation and dictionary learning that uses two nonparametric priors for sparse signal representations: the beta process and the Dirichlet process. Additionally, the model uses spatial information within the image to encourage sharing of information within image subregions. We derive a hybrid MAP/Gibbs sampler, which performs Gibbs sampling for the latent indicator variables and MAP estimation for all other parameters. We present experimental results, where we show an improvement over other state-of-the-art algorithms in the low-measurement regime.

***Index Terms*—** Bayesian models, Dirichlet process, beta process, image interpolation, dictionary learning

1. INTRODUCTION

Nonparametric Bayesian analysis provides a powerful set of tools for modeling data, and has found extensive use in recent research (e.g., [11][12] and references therein). A key advantage of these methods is the sparsity-promotion of the various nonparametric priors, which allows for many truncation issues to essentially be avoided. For example, the Dirichlet process [4] has been useful for uncovering, or inferring the number of components in a mixture model, while the beta process [5] has recently found significant use for inferring the number of factors in latent factor models (see [9]).

In this paper we present a Bayesian nonparametric algorithm for interpolating missing voxel values in incomplete images. The model uses the Dirichlet process and the beta process, as well as spatial information of pixel location within the image. We present results on complex, canonical images employed in the image processing community [6], and show an improvement in performance for high percentages of missing voxels, as well as the advantage provided by all three aspects of the proposed prior. Although we present results on color images, the proposed algorithm is equally applicable to other image types, such as hyperspectral images, where spatial information is still meaningful.

We present and discuss the model in Section 2 and inference equations in Section 3. We show experimental results in Section 4 and conclude in Section 5.

2. THE MODEL

We first describe the model, followed by the hierarchical generative structure. We then discuss the handling of missing data. Let $\{y_n\}_{n=1}^N$ be a collection of N patches of size $m \times m \times 3$ extracted from a color image and reshaped into $P = 3m^2$ dimensional vectors. Also, let $\{x_n\}_{n=1}^N$ be the two-dimensional coordinates for the corresponding patches (e.g., the center pixel location of the patch).

We model each patch, y_n, as a sparse, weighted combination of a dictionary matrix, $\Phi \in \mathbb{R}^{P\times K}$, with additive noise; and each patch location, x_n, as being generated from a mixture of Gaussians, G_d. Above this in the hierarchy is another mixture model, to each component of which belongs a K-dimensional vector, π_d, whose elements define the probability of using their respective vectors (columns) in the dictionary. Also to each component are parameters for the Gaussian mixture model G_d, which governs the spatial locations to which that vector applies.

With reference to (1) below, the Dirichlet process is the prior on the D mixing weights, $\eta \in \Delta_D$, which are probabilities of using a particular component, (i.e., $\{\pi_d, G_d\}$ pair); thus there are two modalities [8]. Though each vector, π_d, is unique, they each correspond to the same dictionary via a hierarchical beta process [11]. The latent indicator, c_n, drawn from η determines the component from which patch y_n and pixel x_n come (i.e., determines that $\{\pi_{c_n}, G_{c_n}\}$ are used). The K-dimensional binary vector, z_n, generated using π_{c_n} then turns on or off dictionary elements for the n^{th} patch, and the weight vector, w_n, provides added flexibility. Drawing pixel locations from a Gaussian mixture model (written GMM$(\cdot)$ for short) imposes that patches that share a component must not only look alike via their usage of the dictionary, but also must be located in the same subregion of the image.

The generative process of the complete (no missing) data set described above is,

$$\begin{aligned} y_n &\sim \mathcal{N}\left(\Phi(w_n \circ z_n), \sigma_\epsilon^2 I\right) \\ x_n &\sim \text{GMM}\left(G_{c_n}\right) \\ w_n(k) &\sim z_n(k)\mathcal{N}\left(0, \sigma_w^2\right) + (1 - z_n(k))\delta_0 \\ z_n(k) &\sim \text{Bernoulli}\left(\pi_{c_n}(k)\right) \\ c_n &\sim \text{Multinomial}\left(\{1, \ldots, D\}, \eta\right) \\ \pi_d(k) &\sim \text{Beta}\left(\frac{\alpha\gamma}{K}, \alpha(1 - \frac{\gamma}{K})\right) \\ \phi_k &\sim \mathcal{N}(0, \sigma_\phi^2 I) \\ G_d &\sim \text{DP}(cG_0) \\ \eta &\sim \text{Dirichlet}\left(\frac{\beta}{D}, \ldots, \frac{\beta}{D}\right) \end{aligned} \tag{1}$$

for $d = 1, \ldots, D$ and $k = 1, \ldots, K$. In the first line, the symbol $\circ$ indicates element-wise multiplication. Also, though the Dirichlet process and beta process are infinite-dimensional priors (i.e., K and D are infinite), we use finite-dimensional approximations, which work well in practice at finding sparse representations when K and D are large.

We finally note the "spike-slab" prior on $w_n(k)$. This is selected over $w_n \sim \mathcal{N}(0, \sigma_w^2 I)$ since it slightly increases the penalty for adding a dictionary element in the inference procedure, and allows for σ_w^2 to be updated using only the active elements of z_n, something we've found important in practice.

2.1. Handling Missing Data

For interpolation and one step of inference, we integrate out the weight vector. We then use two properties of multivariate Gaussian distributions to handle the missing data. To review, if $y \sim \mathcal{N}(\Phi(w \circ z), \sigma_\epsilon^2 I)$ and $w \sim \mathcal{N}(0, \sigma_w^2 I)$, then integrating out w results in $y \sim \mathcal{N}(0, \sigma_\epsilon^2 I + \sigma_w^2 \Phi \text{diag}(z) \Phi^T)$, where $\text{diag}(z)$ forms a diagonal matrix using the vector z; results for the spike-slab prior employed above are the same. Partitioning the vector y and the covariance matrix into their missing and observed parts, generically written as

$$\begin{bmatrix} y_\text{m} \\ y_\text{o} \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \Sigma_\text{m} & \Sigma_\text{m,o} \\ \Sigma_\text{o,m} & \Sigma_\text{o} \end{bmatrix}\right) \tag{2}$$

then integrating out the missing data produces $y_\text{o} \sim \mathcal{N}(0, \Sigma_\text{o})$, which is the distribution used in likelihood calculations. For interpolation, the distribution of y_m given y_o is,

$$y_\text{m} | y_\text{o} \sim \mathcal{N}(\Sigma_\text{m,o}\Sigma_\text{o}^{-1} y_\text{o}, \Sigma_\text{m} - \Sigma_\text{m,o}\Sigma_\text{o}^{-1}\Sigma_\text{o,m})$$

These two properties are applied to the relevant partitions of y_n and $\Sigma_n := \sigma_\epsilon^2 I + \sigma_w^2 \Phi \text{diag}(z_n) \Phi^T$.

To fix notation in what follows, let Y be the $P \times N$ matrix formed by combining all y_n vectors. We define the set I_n^c containing the indices of measured values for the n^th column of Y and similarly define I_p^r for the p^th row of Y. For vectors, $v(I^c)$ selects dimensions of v, while for matrices, A_{I^c, I^r} selects rows and columns of A.

3. MODEL INFERENCE

For model inference, we use both MAP estimation and Gibbs sampling;[1] we sample the latent component indicators, c_n, and latent binary indicators, z_n, and perform MAP updates for the parameters Φ and w_n. When sampling c_n and z_n, we integrate out, or marginalize the values of the mixing weights η and the latent factor probabilities π_d. We also perform MAP inference to learn all variance parameters using conjugate inverse-gamma priors, which we omit for space. These variance parameters are the Bayesian equivalent of regularization terms for optimization, and inference for them significantly improves the performance of the model.

3.1. Maximum A Posteriori Updates

MAP update for w_n:

$$w_n = \left(\frac{\sigma_\epsilon^2}{\sigma_w^2} I + \Phi_{I_n^c,:}^T \Phi_{I_n^c,:} \circ z_n z_n^T\right)^{-1} \text{diag}(z_n) \Phi_{I_n^c,:}^T y_n(I_n^c) \tag{3}$$

These updates are ℓ_2-regularized least squares solutions [2] calculated using only the activated dictionary elements for the current observation, as indicated by z_n.

MAP update for Φ: We define the matrix W in a similar manner as Y in Section 2.1. The p^th dimension of the updated dictionary is,

$$\Phi_{p,:} = Y_{p,I_p^r} W_{:,I_p^r}^T \left(\frac{\sigma_\epsilon^2}{\sigma_\phi^2} I + W_{:,I_p^r} W_{:,I_p^r}^T\right)^{-1} \tag{4}$$

We note that W will have zeros in the same locations as Z, and hence we do not have to write $W \circ Z$. The diagonal prior covariance in the dictionary allows for this analytical result.

Marginalizing π_d: We integrate out the values of π_d to obtain the vector denoted $\hat{\pi}_d$,

$$\hat{\pi}_d(k) = \frac{\frac{\alpha\gamma}{K} + \sum_{n=1}^N z_n(k) \mathbb{I}(c_n = d)}{\alpha + \sum_{n=1}^N \mathbb{I}(c_n = d)} \tag{5}$$

for $k = 1, \ldots, K$.

Marginalizing η: We integrate out the mixing weights to obtain the vector denoted $\hat{\eta}$,

$$\hat{\eta}(d) = \frac{\frac{\beta}{D} + \sum_{n=1}^N \mathbb{I}(c_n = d)}{\beta + N} \tag{6}$$

for $d = 1, \ldots, D$.

MAP update for G_d: The mixing weights, means and covariances of G_d are calculated using all x_n for which the indicator $c_n = d$. We use a finite-dimensional approximation to the DP; update equations can be found in [2].

[1] Code is available at www.ee.duke.edu/∼jwp4/ICIP2010

3.2. Gibbs Sampling of Latent Indicators

Sample c_n: The latent component indicator is sampled from a D-dimensional multinomial distribution, with

$$p(c_n = d|\Theta) \propto p(z_n|\hat{\pi}_d)p(x_n|G_d)p(c_n = d|\hat{\eta}) \quad (7)$$

where $p(z_n|\hat{\pi}_d) = \prod_{k=1}^{K} \hat{\pi}_d(k)^{z_n(k)}(1 - \hat{\pi}_d(k))^{1-z_n(k)}$, $p(c_n = d|\hat{\eta}) = \hat{\eta}(d)$ and $p(x_n|G_d)$ is the likelihood calculated using the GMM G_d. The symbol Θ represents the set of all parameters and latent indicators.

Sample z_n: For the sampling of the latent binary indicators, we integrate out the corresponding weights, w_n, in the way discussed in Section 2.1. Let $\tilde{z}_n$ be the binary indicator vector of the previous iteration. Using the definitions

$$M_n := \sigma_\epsilon^2 I + \sigma_w^2 \Phi_{I_n^c,:} \operatorname{diag}(\tilde{z}_n) \Phi_{I_n^c,:}^T$$

$$\xi_n^k := \sigma_w^2 \phi_k(I_n^c)^T M_n^{-1} \phi_k(I_n^c)$$

then,

$$\ln p(z_n(k) = 1|\Theta) \propto \ln \hat{\pi}_{c_n}(k) \; - \; \mathbb{I}\left(\tilde{z}_n(k) = 0\right) \times \frac{1}{2}\left(\ln(1 + \xi_n^k) - \frac{\sigma_w^2(\phi_k(I_n^c)^T M_n^{-1} y_n(I_n^c))^2}{1 + \xi_n^k} + \ln 2\pi\sigma_w^2\right) \quad (8)$$

$$\ln p(z_n(k) = 0|\Theta) \propto \ln\left(1 - \hat{\pi}_{c_n}(k)\right) - \mathbb{I}\left(\tilde{z}_n(k) = 1\right) \times \frac{1}{2}\left(\ln(1 - \xi_n^k) + \frac{\sigma_w^2(\phi_k(I_n^c)^T M_n^{-1} y_n(I_n^c))^2}{1 - \xi_n^k} - \ln 2\pi\sigma_w^2\right) \quad (9)$$

One of the two indicators will be active, which accounts for the effect of either the addition or subtraction of both a dictionary element and a $w_n(k)$ term ($\pi = 3.141...$ in $\ln 2\pi\sigma_w^2$). We do not update the vector $\tilde{z}_n$ element by element, but all K dimensions at once to save computation time. We note that the matrix inversion lemma and a property of matrix determinants were used in this derivation.

4. EXPERIMENTS

We first show experimental results for the image in Figure 1, also used in [6]. We extracted $5 \times 5 \times 3$ overlapping patches from the image centered on each pixel for which the patch does not fall outside the image. No prior training is performed on separate images to aid inference, as is done in [6], but rather all learning is done *in situ*. We also considered other patch sizes, for example $7 \times 7 \times 3$ (reconstruction PSNR $= 29.25$, compared with PSNR $= 29.65$ in [6], where prior training was done), and $8 \times 8 \times 3$ (PSNR $= 29.47$, compared with PSNR $= 29.31$ in [12], where Gibbs sampling was performed throughout), these PSNR values being for 80% of the voxels missing at random. As our intention is to compare performance between algorithms, we present results for $5 \times 5 \times 3$, noting that similar results were observed for other patch sizes.

We compare with five other algorithms in Figure 3: 1) The proposed model *without* spatial information, 2) The model in [12], which is the proposed model without the Dirichlet process or spatial information, 3) the K-SVD algorithm [1] *without* using a prior database, 4) the MOD algorithm [3] and 5) an iterative minimum MSE (iMMSE) algorithm [7] in which the dictionary size increases one element at a time, followed by a minimization of the squared error to all measured values.

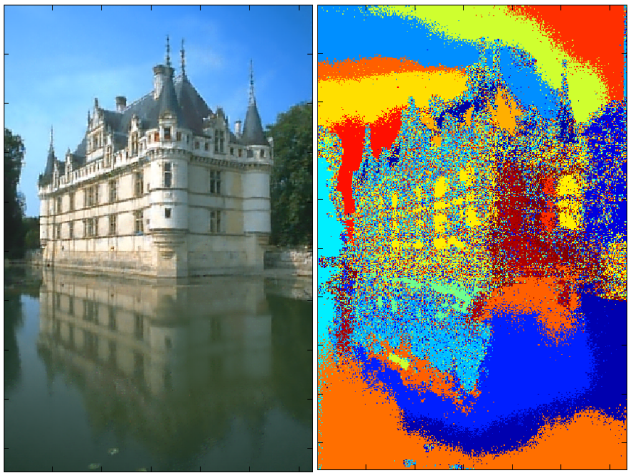
The K-SVD and MOD algorithms make extensive use of the OMP algorithm [10], which requires a sparsity setting, T, determining the number of dictionary elements to be used for each patch. Defining p to be the probability of a missing pixel, we set this value to $T \approx (1-p)3m^2/2$, which is half of the expected number of measured voxels in a patch. For 80% missing, this value is $T = 7$. We ran 200 iterations of each algorithm (except iMMSE), which was sufficient to converge to a stable PSNR. For iMMSE, the stopping point (rank of the factorized matrix) can be determined by setting a threshold on the approximation error, or by viewing the image; we simply increase the dictionary and use the best PSNR result, which is not practical, but does give an empirical upper bound on performance.

In Figures 1 and 2, we present example reconstruction results, where the clustering clearly shows the spatial aspect of the prior; see [6] for original images. This helps explain the advantage of the model in the low-measurement regime, shown in Figure 3 for the castle image; patches with few measurements may have difficulty clustering based only upon their dictionary usage, and spatial information can improve this clustering by encouraging patches to cluster by region as well as appearance. Separating patches by region makes the usage of the dictionary less ambiguous by allowing patches to more effectively share statistical strength, which in turn aids in constructing a better dictionary. In Table 1, we show the cost of this added complexity in runtime. Similar performance was observed for other images, which we omit for space. We note that in Figure 3, the PSNR is calculated using only the missing voxels.

5. CONCLUSION

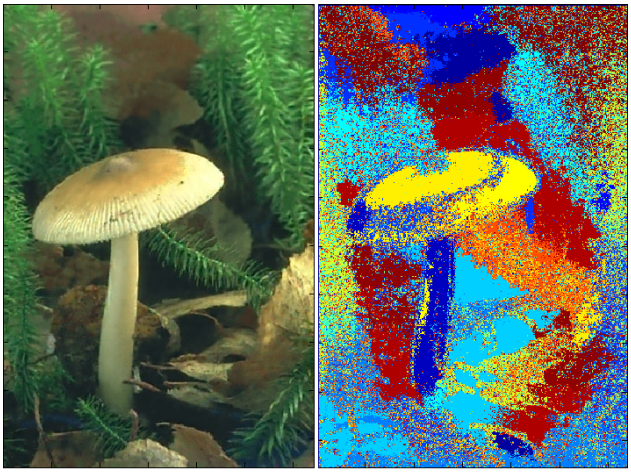
We have presented a Bayesian nonparametric model for image interpolation. The model uses the beta process for dictionary learning, the Dirichlet process for flexibility in dictionary usage and spatial information within the image, which encourages similar dictionary usage within subregions of the image. Experiments on several images (only two shown due to space) showed an advantage of our model compared with other algorithms in the low-measurement regime.

In future work, we plan to apply this model to hyperspectral images, which requires no alteration to the model or inference procedure. This could help significantly reduce the number of required measurements, and may be an alternative to other compressive sensing approaches (see [12]).



(a) (b)

Fig. 1. Example result (80% random missing, 5×5×3 patch): (a) Reconstructed image: PSNR = 28.76 (b) Clustering results: Cluster index as a function of pixel location.



(a) (b)

Fig. 2. Example result (80% random missing, 5×5×3 patch): (a) Reconstructed image: PSNR = 29.73 (b) Clustering results: Cluster index as a function of pixel location.

	Time per iteration (minutes)			
	60%	70%	80%	90%
BP & DP & Spatial	3.52	3.04	2.67	2.30
BP & DP, No Spatial	2.23	1.81	1.38	0.96
BP Only	2.02	1.63	1.26	1.15
K-SVD	2.50	1.64	1.11	0.69
MOD	2.58	1.73	1.17	0.67

Table 1. Average per-iteration run time for algorithms as function of percent missing data. Comparison is not meaningful for the iMMSE algorithm (which is significantly faster).

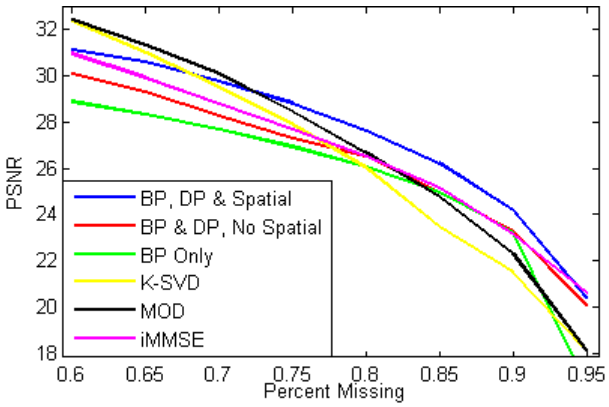


Fig. 3. Castle image: PSNR of interpolated missing data using $5 \times 5 \times 3$ patches averaged over five trials. The proposed algorithm performs well for low-measurement percentages. We set $K = 100$ and $D = 50$.

6. REFERENCES

[1] M. Aharon, M. Elad and A. Bruckstein (2006). K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans on Sig Proc*, 54(11): 4311-4322.

[2] C.M. Bishop (2006). *Pattern Recognition and Machine Learning*, Springer, New York.

[3] K. Engan, S.O. Aase and J.H. Huszy (1999). Method of optimal directions for frame design. *Proc. of ICASSP*, 5:2443-2446.

[4] T. Ferguson (1973). A Bayesian analysis of some nonparametric problems. *Annals of Statistics*, 1:209-230.

[5] N.L. Hjort (1990). Nonparametric Bayes estimators based on beta processes in models for life history data. *Annals of Statistics*, 18(3):1259-1294.

[6] J. Mairal, M. Elad and G. Sapiro (2008). Sparse representation for color image restoration. *IEEE Trans. Image Proc.*, vol. 17.

[7] J. Nocedal and S.J. Wright (2006). *Numerical Optimization, Second Edition*, Springer, New York.

[8] J. Paisley and L. Carin (2009). Dirichlet process mixture models with multiple modalities. *Proc. of ICASSP*, pp. 1613-1616.

[9] J. Paisley and L. Carin (2009). Nonparametric factor analysis with beta process priors. *Proc. of ICML*, pp. 777-784.

[10] Y.C. Pati, R. Rezaiifar, P.S. Krishnaprasad (1993). Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. *Asilomar Conf on Signals Systems and Computers*.

[11] R. Thibaux and M.I. Jordan (2007). Hierarchical beta processes and the Indian buffet process. *AISTAT 2007*.

[12] M. Zhou, H. Chen, J. Paisley, L. Ren, G. Sapiro and L. Carin (2009). Non-parametric Bayesian dictionary learning for sparse image representations. *NIPS 2009*.