Beta Process Joint Dictionary Learning for Coupled Feature Spaces with Application to Single Image Super-Resolution

Li He, Hairong Qi, Russell Zaretzki
The University of Tennessee, Knoxville
{lhe4,hqi,rzaretzk}@utk.edu

Abstract

This paper addresses the problem of learning over-complete dictionaries for the coupled feature spaces, where the learned dictionaries also reflect the relationship between the two spaces. A Bayesian method using a beta process prior is applied to learn the over-complete dictionaries. Compared to previous couple feature spaces dictionary learning algorithms, our algorithm not only provides dictionaries that customized to each feature space, but also adds more consistent and accurate mapping between the two feature spaces. This is due to the unique property of the beta process model that the sparse representation can be decomposed to values and dictionary atom indicators. The proposed algorithm is able to learn sparse representations that correspond to the same dictionary atoms with the same sparsity but different values in coupled feature spaces, thus bringing consistent and accurate mapping between coupled feature spaces. Another advantage of the proposed method is that the number of dictionary atoms and their relative importance may be inferred non-parametrically. We compare the proposed approach to several state-of-the-art dictionary learning methods by applying this method to single image super-resolution. The experimental results show that dictionaries learned by our method produces the best super-resolution results compared to other state-of-the-art methods.

1. Introduction

The use of over-complete dictionaries for sparse representation has been the subject of extensive research over the last decade. Research on signal processing [13] suggests that over-complete bases offer the flexibility to represent much wider range of signals with more elementary basis atoms than the signal dimension. Research on image statistics [15, 16] suggests that image patches can be well represented as a sparse linear combination of elements from an appropriately chosen over-complete dictionary. There have been numerous methods proposed to design such over-complete dictionaries [1, 9, 12, 14, 17, 19, 21]. Dictionaries learned by these methods yield sparse representations that have higher recovery accuracy than do with conventional representations, therefore attaining state-of-the-art performances on denoising, in-painting, image abstraction and super-resolution.

In many signal processing problems, we have coupled feature spaces, e.g., the image patch space and sketch patch space for photo-sketch abstraction, the original and compressed signal spaces in compressive sensing, and the high-resolution patch space and low-resolution patch space in patch-based image super-resolution. The intuitive method to learn dictionaries for coupled feature spaces is using single sparse coding model to learn the coupled dictionaries in concatenated spaces [25]. However, dictionaries learned this way usually cannot capture the complex, spatial-variant and nonlinear relationship between the two feature spaces.

Several algorithms have been proposed to solve this problem [22, 24, 27]. Zeyde [27] *et al*. proposed a two-step learning algorithm, where one dictionary is learned by KSVD [1] and the other is generated via least-square. Although the dictionaries are learned individually, same coefficients are still used for the two feature spaces, limiting the dictionaries from being customized to both spaces. Wang [22] proposed a semi-coupled training model to solve the problem where a mapping matrix is used to capture the relationship of the sparse representations between spaces. Although the learned dictionaries can better minimize the error in both spaces than those learned in concatenated spaces, the corresponding relationship of dictionaries in the two feature spaces are not captured during the learning process. Yang [24] provided a bilevel optimization solution of the problem. Instead of solving the two optimization problems in two feature spaces together [26], the bilevel method moves one of the optimization problem to the regularization term of the other problem. Although the learned sparse representation of bilevel method has less learning errors, the same sparse coding is still required for both feature spaces.

In this paper, a beta process joint dictionary learning

(BP-JDL) algorithm is proposed for dictionary learning problems in coupled feature spaces. Recent research on using non-parametric Bayesian approach [6, 17] to learn an over-complete dictionary offers several advantages not found in earlier approaches and shows significant improvement in applications such as image denoising, inpainting and compressive sensing [28]. However, those approaches are only suitable for dictionary learning in single feature space. We propose a new beta process model which is customized for the problem of learning dictionaries in coupled feature spaces. Our model, together with [22, 24], provides dictionary learning methods that customized to each feature space, however, our method adds more consistent and accurate mapping between the two feature spaces. This is due to the unique property of the beta process model [17] that the sparse representations can be decomposed to values and dictionary atom indicators. We use the same beta process prior for dictionary atom indicators but different priors for values in two feature spaces. In this way, the proposed algorithm is able to learn sparse representations that correspond to the *same* dictionary atoms with the *same* sparsity but *different* values in coupled feature spaces, thus bringing consistent and accurate mapping between coupled feature spaces. In addition, in previous over-complete dictionary learning methods, the dictionary size is an unknown parameter and a large-size dictionary is necessary to produce good recovery accuracy. BP-JDL may infer dictionary size non-parametrically and produce the same or better learning accuracies with much smaller dictionary size.

In order to compare BP-JDL with state-of-the-art coupled feature space dictionary learning methods, we tailor BP-JDL to the dictionary learning problem of the patch-based single image super-resolution. Experimental results show that BP-JDL outperforms previous methods in terms of both quality of super-resolution and recover accuracy.

The rest of the paper is organized as follows. Section 2 describes the related work. Section 3 describes the beta process joint dictionary learning for coupled feature spaces. Section 4 demonstrates results of the single image super-resolution application. Section 5 concludes experiments and discusses future work.

2. Related Works

Many image analysis problems use coupled feature spaces [7, 10, 11, 25]. In this paper, we focus on the problem of patch-based single image super-resolution, since several dictionary learning algorithms have been proposed for this application.

Super-resolution is a technique that enhances the resolution of an image or multiple images of the same scene. The recently studied single image super-resolution (SISR) problem attempts to enhance the resolution of a single image via offline learned patch-based dictionaries. The low-resolution (low-res) image is down-sampled from a blurred high-resolution (high-res) image and often the blurring kernel is unknown. Many methods [4, 20, 22, 24, 25, 27] have been proposed trying to capture the concurrent prior between the low- and high-resolution patches using dictionary learning techniques. In these methods, a high-res patch is normally recovered using the high-res dictionary and sparse coefficients calculated using the low-res feature patch and low-res feature dictionary. Therefore, we need to learn these two dictionaries in both high-res and low-res feature spaces. This is a typical dictionary learning problem in coupled feature spaces.

The first approach that generated the state-of-the-art SISR result concatenates the two feature spaces together, thus converting the problem to dictionary learning in single feature space. Since the learning of an over-complete dictionary is often an NP-hard problem, many approximation algorithms have been proposed, such as RVM [21], KSVD [1], online dictionary learning [12], efficient sparse coding [9], and beta process [17]. All these methods are able to generate the over-complete dictionary and sparse coefficients. Once the dictionaries are learned, we can use one dictionary to calculate the sparse coefficients and the other dictionary to recover the desired signal. However, because the sparse coefficients are shared between the two dictionaries, the algorithm normally finds it difficult to fit the dictionary and coefficients to both feature spaces. Therefore, a further learning model is necessary to adapt the dictionary learning algorithm to coupled feature spaces.

The second approach is to learn dictionary from one space first then generate the other dictionary via least square. Zeyde [27] used this approach for the SISR problem, where the low-res dictionary is learned and the high-res dictionary is generated via least square. Although this method largely decreases the computational cost because only one dictionary is learned and the dictionary is well-fitted in the low-res patch space, the same is not true in the high-res patch space. A simultaneous dictionary learning algorithm is thus essential to balance the learning errors in both feature spaces.

The most recent approaches, also referred to as the semi-coupled approaches [22, 24], seek to improve the learning result by letting the dictionaries fit the two feature spaces better. Yang [24] formulated the problem as a bilevel optimization problem while Wang [22] used a mapping function to characterize the relationship of the two feature spaces. Yang's method still shares the coefficients between the two feature spaces and both methods did not enforce the corresponding relationship between the learned dictionaries. We resolve these two issues by taking advantage of the beta process prior model.

Recent non-parametric Bayesian approaches such as the Indian Buffet Process (IBP) [6] and the beta process

(BP) [17] for latent factor analysis have been extensively studied. BP is more suitable for dictionary learning compared to IBP because it has more flexibility. However, BP is developed to learn dictionary in single feature space and may not be suitable to learn dictionaries in coupled feature spaces. Nevertheless, the truncated beta process allows the sparse coefficients to be expressed as an element-wise multiplication of a *binary* latent factor indicator and a *normal* coefficient value. We can take advantage of this property in the dictionary learning problem of coupled feature spaces by restraining the coefficients in coupled feature spaces to use the same dictionary atom indicator but different coefficient values.

In addition, the desired sparsity property of the dictionary coefficients, used in previous over-complete dictionary learning methods, can be naturally incorporated in the beta process, thus allowing dictionary size to tend to infinity while the training samples only use a small subset of dictionary atoms via the sparse coefficients. Finally, in many applications, the dictionary size and the desired sparsity level need to be manually set [1,26]. However, these two parameters are better to be inferred automatically. There have been recent interests in applying non-parametric Bayesian methods [8,18] to infer the number of dictionary atoms based on the observed data. A Bayesian approach proposed in [17] provided a solution for this problem.

3. Beta Process Joint Dictionary Learning for Coupled Feature Space

Suppose we have two coupled feature spaces $\mathcal{Y} \in \mathbb{R}^{P_y}$ and $\mathcal{X} \in \mathbb{R}^{P_x}$, where the features are sparse in terms of certain dictionaries. There exists a mapping function $\mathcal{F}: \mathcal{Y} \to \mathcal{X}$ that relates features in $\mathcal{Y}$ to the corresponding features in $\mathcal{X}$. Therefore, the relation of the dictionaries and the observations and the relation of the two feature spaces can be described as

$$\begin{aligned} \mathbf{x}_i &= \mathbf{D}^{(x)}\alpha_i^{(x)} + \epsilon_i^{(x)} \\ \mathbf{y}_i &= \mathbf{D}^{(y)}\alpha_i^{(y)} + \epsilon_i^{(y)} \\ \mathbf{M}\alpha_i^{(y)} &= \alpha_i^{(x)} \end{aligned} \tag{1}$$

where $\mathbf{x}_i, \mathbf{y}_i, i = 1, \ldots, N$ are training samples with dimensions P_x and P_y, respectively. $\mathbf{D}^{(x)} = (\mathbf{d}_1^{(x)}, \mathbf{d}_2^{(x)}, \ldots, \mathbf{d}_K^{(x)})$ and $\mathbf{D}^{(y)} = (\mathbf{d}_1^{(y)}, \mathbf{d}_2^{(y)}, \ldots, \mathbf{d}_K^{(y)})$ are dictionaries learned in each space and both dictionaries have K atoms. $\alpha_i^{(x)}$ and $\alpha_i^{(y)}$ are coefficients of each dictionary. $\epsilon_i^{(x)}$ and $\epsilon_i^{(x)}$ are the recovery errors. M is a mapping matrix from sparse coding of $\mathbf{y}_i$ to $\mathbf{x}_i$. In order to learn two dictionaries at the same time, previous algorithms [24, 26] use the same coefficients for both dictionaries, i.e., $\alpha_i^{(x)} = \alpha_i^{(y)}$. In this way, one might concatenate two feature spaces and convert the dictionary learning problem of coupled feature spaces to the dictionary learning problem of single feature space. However, allowing different coefficients in two feature spaces provides a better fitting of learning and the learned dictionaries are more customized to individual feature space. Beta process [17] allows the decomposition of the coefficients to the element multiplication of dictionary atom indicators and coefficient values, providing the much needed flexibility to fit each feature space better while still maintaining the correspondence between the two dictionaries.

We develop a new beta process based on [28] to tackle the dictionary learning problem in coupled feature spaces. The new two-parameter beta process with parameters $a, b > 0$ and base measure H_0, is represented as $BP(a, b, H_0)$ and may be written in set function form as

$$\begin{aligned} H &= \sum_{k=1}^{K} \pi_k \delta_{\mathbf{d}_k^{(x)}} = \sum_{k=1}^{K} \pi_k \delta_{\mathbf{d}_k^{(y)}} \\ \pi_k &\sim Beta(a/K, b(K-1)/K), \quad \mathbf{d}_k^{(x)}, \mathbf{d}_k^{(y)} \sim H_0 \end{aligned} \tag{2}$$

where $\delta_{\mathbf{d}_k^{(x)}}$ and $\delta_{\mathbf{d}_k^{(y)}}$ are unit point mass at $\mathbf{d}_k^{(x)}$ and $\mathbf{d}_k^{(y)}$. We use a single beta process prior and the same dictionary atom indicator to connect the two feature spaces. π_k represents a vector of K probabilities, each associated with the respective atom $\mathbf{d}_k^{(y)}$ and the corresponding $\mathbf{d}_k^{(x)}$. H is composed by infinite number of $\mathbf{d}_k^{(y)}$ (as well as $\mathbf{d}_k^{(x)}$) sampled from H_0 and is a valid measure when $K \to \infty$. A finite approximation of H can be made by simply setting K to a large, but finite number.

Following the general structure of beta process described in [28], the beta process joint dictionary learning model for the coupled feature spaces may be expressed as

$$\begin{aligned} \mathbf{x}_i &= \mathbf{D}^{(x)}\alpha_i^{(x)} + \epsilon_i^{(x)}, & \mathbf{y}_i &= \mathbf{D}^{(y)}\alpha_i^{(y)} + \epsilon_i^{(y)} \\ \alpha_i^{(x)} &= \mathbf{z}_i \circ \mathbf{s}_i^{(x)}, & \alpha_i^{(y)} &= \mathbf{z}_i \circ \mathbf{s}_i^{(y)} \\ \mathbf{d}_k^{(x)} &\sim N(0, P_x^{-1}\mathbf{I}_{P_x}), & \mathbf{d}_k^{(y)} &\sim N(0, P_y^{-1}\mathbf{I}_{P_y}) \\ \mathbf{s}_i^{(x)} &\sim N(0, \gamma_{s^{(x)}}^{-1}\mathbf{I}_K), & \mathbf{s}_i^{(y)} &\sim N(0, \gamma_{s^{(y)}}^{-1}\mathbf{I}_K) \\ \mathbf{z}_i &\sim \prod_{k=1}^{K} Bernoulli(\pi_k), & \pi_k &\sim Beta(a/K, b(K-1)/K) \\ \epsilon_i^{(x)} &\sim N(0, \gamma_{\epsilon^{(x)}}^{-1}\mathbf{I}_{P_x}), & \epsilon_i^{(y)} &\sim N(0, \gamma_{\epsilon^{(y)}}^{-1}\mathbf{I}_{P_y}) \\ \gamma_{s^{(x)}}, \gamma_{s^{(y)}} &\sim \Gamma(c, d), & \gamma_{\epsilon^{(x)}}, \gamma_{\epsilon^{(y)}} &\sim \Gamma(e, f) \end{aligned} \tag{3}$$

In order to constrain that $\mathbf{x}_i$ uses the same corresponding dictionary atom as that used by $\mathbf{y}_i$, we choose the same dictionary atom indicator $\mathbf{z}_i$ for both $\mathbf{d}_k^{(x)}$ and $\mathbf{d}_k^{(y)}$. At the same time, in order to provide different coefficient values, weights $\mathbf{s}_i^{(x)}$ and $\mathbf{s}_i^{(y)}$ are drawn from different distributions, as part of the coefficients. Finally we have the coefficients

$\alpha_i^{(x)} = \mathbf{z}_i \circ \mathbf{s}_i^{(x)}$ and $\alpha_i^{(y)} = \mathbf{z}_i \circ \mathbf{s}_i^{(y)}$, where $\circ$ is an element-wise multiplication. Because $\alpha^{(y)}$ and $\alpha^{(x)}$ use the same dictionary atom indicator $\mathbf{z}_i$, they have the same number of non-zero elements and the corresponding relationship of dictionary atoms in the two feature spaces are enforced during the learning process.

Specifically, N *binary* vectors $\mathbf{z}_i \in \{0,1\}^K, i = 1,\ldots,N$ are drawn from H and the kth component of $\mathbf{z}_i$ is drawn from $z_{ik} \sim Bernoulli(\pi_k)$. These N binary column vectors are used to constitute the dictionary atom indicator matrix $\mathbf{Z} \in \{0,1\}^{K\times N}$, with the ith column corresponding to $\mathbf{z}_i$ and the kth row associated with both $\mathbf{d}_k^{(x)}$ and $\mathbf{d}_k^{(y)}$. Next, weights $\mathbf{s}_i^{(x)} \sim N(0, \gamma_{s^{(x)}}^{-1}\mathbf{I}_K)$ are drawn as part of the coefficients. $\mathbf{I}_K$ is an identity matrix indicating that we use the same $\gamma_{s^{(x)}}^{-1}$ for all $(s_{i1}^{(x)} \ldots s_{iK}^{(x)})$. The $\circ$ in $\alpha_i^{(x)} = \mathbf{z}_i \circ \mathbf{s}_i^{(x)}$ represents element-wise multiplication of two vectors. Weights $\mathbf{s}_i^{(y)}$ are drawn in the similar way.

For the purpose of building a fully conjugate model, the dictionary atoms $\mathbf{d}^{(\mathbf{x})}{}_k$ are drawn from a multivariate zero-mean Gaussian (H_0) with variance $P_x^{-1}\mathbf{I}_{P_x}$ and the error vectors $\epsilon^{(\mathbf{x})}{}_i$ are drawn from a zero-mean Gaussian with variance $\gamma_{\epsilon^{(x)}}^{-1}\mathbf{I}_P$. In addition, because the inverse Gamma distribution is conjugate with the Gaussian distribution, $\gamma_{s^{(x)}}$ are drawn from the Gamma distributions. The non-informative Gamma hyper-prior is placed on $\gamma_{s^{(x)}}$, where we initialize $c = d = 10^{-6}$. We also apply the same distribution to $\mathbf{d}_k^{(y)}$, $\epsilon_i^{(y)}$, $\gamma_{s^{(y)}}$ and $\gamma_{\epsilon^{(y)}}$. In this model, the expected sparsity level in a training sample $\mathbf{x}_i$ or $\mathbf{y}_i$ as $K \to \infty$ is drawn from $Poisson(a/b)$. We set $a = b = 1$, but one may change values of a and b. However, [28] proved the sparsity level is not sensitive to different values of a and b and is intrinsic to the data. Finally, after we learned $\alpha^{(y)}$ and $\alpha^{(x)}$, the mapping matrix $\mathbf{M}$ can be calculated via the least square:

$$\mathbf{M} = [(\alpha^{(y)}\alpha^{(y)^T})^{-1}\alpha^{(y)}\alpha^{(x)^T}]^T \tag{4}$$

Elements in Eq. 3 are in the conjugate exponential family, and therefore the posterior inference may be implemented via Gibbs-sampling method with analytic update equations. The Gibbs sampling update equations can be found in Appendix A.

4. Single Image Super-Resolution Application

The single image super-resolution (SISR) asks to recover the high-res image ($\mathbf{H}$) from a low-res image ($\mathbf{L}$), with the observation model expressed as: $\downarrow B\mathbf{H} = \mathbf{L}$, where $\downarrow$ is a downsample operator and B is a blur operator. With an input low-res image, the SISR problem asks to recover the high-res image by reversing the process of downsample and blur. Instead of reversing the process directly, Yang [25] suggested that we can use learned dictionaries of high-res feature space and low-res feature space to reconstruct the high-res image. The two feature spaces are constructed as:

$$\mathbf{x}_i = h; \mathbf{y}_i = [F_1 l; F_2 l; F_3 l; F_4 l] \tag{5}$$

where h is a high-res patch and l is a low-res patch. $F_1 \ldots F_4$ are four (linear) feature extraction operators which are used to penalize visually salient high-frequency errors: $F_1 = [-1, 0, 1], F_2 = F_1^T, F_3 = [1, 0, -2, 0, 1], F_4 = F_3^T$.

We use the proposed BP-JDL method to learn $\mathbf{D}^{(x)}$, $\mathbf{D}^{(y)}$ and the mapping matrix $\mathbf{M}$ for the two feature spaces. Once the dictionaries are learned, we can use them for super-resolution reconstruction. The single image super-resolution reconstruction can be carried out in four steps. The first step calculates the sparse coding of observed low-res feature using learned low-res feature dictionary. In order to compare our dictionary with dictionaries learned by [22,24,26], we use the standard ℓ_1 sparse coding method for step 1 [9]. The second step maps the sparse coding of the low-res feature to sparse coding of the high-res feature using the learned matrix $\mathbf{M}$. The third step recovers the high-res patch using the learned high-res feature dictionary. Because we do not directly use the low-res patch in Eq. 5, the reconstructed high-res image $\mathbf{H}_0$ may not satisfy the constraint $\downarrow B\mathbf{H} = \mathbf{L}$, thus the last step enforces a global constraint to eliminate this inconsistency by projecting $\mathbf{H}_0$ onto the solution space of $\downarrow B\mathbf{H} = \mathbf{L}$. In addition, because the recently introduced non-local redundancies in image are useful for image restoration [2, 5], we also incorporate the non-local self-similarities in step 4. The four steps are summarized in Algorithm 1.

Eq. 9 can be solved by back projection method introduced in [3].

4.1. Experimental Design

We evaluate the performance of the proposed BP-JDL method when applied to single image super-resolution from perspectives of both the quality and the fidelity of the high-resolution image.

Dictionaries for factors of 2 and 3 magnification are learned and used for generating super-resolution images. The low-resolution patches are upsampled to the same size as the high-resolution patches. All dictionaries are trained from 100,000 patch pairs sampled from 10 category representative and texture rich images. The patch pairs are only sampled from the luminance channel of the training images because human eyes are more sensitive to luminance changes. We set the initial dictionary size K of BP-JDL as 1024, 2048 and 4096 to test the capability of BP-JDL's K inference. We use 10000 Gibbs samples for BP-JDL, where the burn-in is 9500 samples and the dictionary is averaged using the rest 500 samples.

Algorithm 1 Single Image Super Resolution

Input: Low-res image $\mathbf{L}$, learned $\mathbf{D}^{(x)}$, $\mathbf{D}^{(y)}$ and $\mathbf{M}$
Output: High-res image $\mathbf{H}^*$
Step 1 Sample low-res patch l_i from the input image $\mathbf{L}$ with overlap ω. Construct $\mathbf{y}_i$ using the four feature extraction operators. Learn $\alpha_i^{(y)}$ using the ℓ_1 sparse coding:

$$\alpha_i^{(y)} = \arg\min_{\alpha_i^{(y)}} \|\mathbf{D}^{(y)}\alpha_i^{(y)} - \mathbf{y}_i\|_2^2 + \lambda\|\alpha_i^{(y)}\|_1 \quad (6)$$

Step 2 Map the sparse coefficients $\alpha^{(y)}$ to $\alpha^{(x)}$ using the learned $\mathbf{M}$:

$$\alpha_i^{(x)} = \mathbf{z}_i\mathbf{M}\alpha_i^{(y)} \quad (7)$$

where $\mathbf{z}_i$ is a binary vector that $z_{ik} = 1$ if $\alpha_{ik}^{(y)} \neq 0$.
Step 3 Recover the high-res patch h_i using $\alpha^{(x)}$ and learned $\mathbf{D}^{(x)}$:

$$h_i = \mathbf{D}^{(x)}\alpha_i^{(x)} \quad (8)$$

After the recovery of all high-res patches, the initial high-res image $\mathbf{H}_0$ can be reconstructed with overlap ω.
Step 4 A global constraint and a non-local similarity constrain are enforced to further improve the reconstruction accuracy:

$$\begin{aligned} &\mathbf{H}^* = \arg\min_{\mathbf{H}} \|\mathbf{H} - \mathbf{H}_0\|^2 \\ &\text{s.t.} \downarrow B\mathbf{H} = \mathbf{L}, \|h_i - \sum_{m=1}^{M} b^m h_{i(0)}^m\|_2^2 \leq \epsilon \end{aligned} \quad (9)$$

where h_i and $h_{i(0)}$ are patches in $\mathbf{H}$ and $\mathbf{H_0}$, respectively. $h_{i(0)}^m$ is the m^{th} most similar patch to $h_{i(0)}$ and b^m is the non-local weight defined in [2].

For the super-resolution reconstruction, high-resolution test images are blurred and down-sampled to $1/4$ and $1/9$ of the original size to produce the input low-resolution images. The high-resolution images are reconstructed using Algorithm 1 with λ set to 0.15 and the overlap set to its max value (i.e., patch size-1). In addition, images reconstructed using the Bicubic interpolation are compared as well.

4.2. Result

4.2.1 Dictionary Learning

Firstly, dictionaries in coupled feature spaces are learned using the proposed BP-JDL algorithm. Compared to the dictionaries learned in concatenated spaces [26], the dictionaries learned by BP-JDL are able to reduce the learning root-mean-square (RMS) errors of high-res feature space $\mathcal{X}$ and low-res feature space $\mathcal{Y}$ by 27.5% and 40.5%, respectively. This result confirms that BP-JDL is capable to learn dictionaries that fit the data better by allowing the different coefficients values for the two spaces.

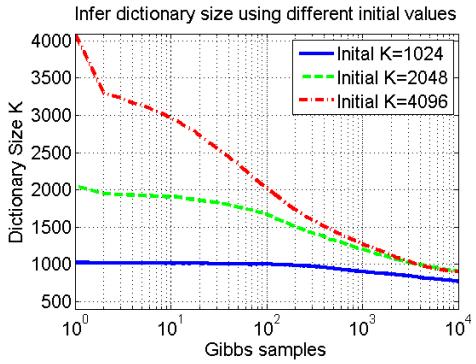

Figure 1. BP-JDL infers dictionary size non-parametrically.

Secondly, the dictionary size inferred by BP-JDL is shown in Figure 1. During the Gibbs sampling process, we search the unused dictionary atoms and delete them. Because BP-JDL has the non-parametric advantage, with different initial Ks, the dictionary size decreases rapidly during the first 1000 samples and gradually converges to similar values, confirming that BP-JDL can infer appropriate dictionary size no matter what the initial value is. With the initial size of 1024, the BP-JDL inferred that $K = 771$ is an appropriate dictionary size. If we fix the dictionary size to 1024 for BP-JDL, the learning RMS errors and sparsity level of the 1024-size dictionaries stay the same as the 771-size dictionaries, indicating that 771 is the appropriate dictionary size for the training data. If the dictionary size is unknown, normally we need exhaustively search for the optimal size. Yang [26] found that the the 1024-size dictionary is optimal, however, the 771-size dictionary may have the same super-resolution performance as the 1024-size dictionary. Besides, since super-resolution using a smaller size dictionary needs less computational power, it may significantly affect the speed and energy consumption of super-resolution applications in resource-constrained environments.

4.2.2 Single Image Super-Resolution

We evaluate the super-resolution (SR) results via peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [23]. Higher SSIM indicates more similar structure between the recovered image and the original image. Factors of 2 and 3 SR results are shown in Tables 1 and 2, respectively. In addition, the visual comparison example of factors of 2 and 3 SR results are shown in Fig. 3 and Fig. 4, respectively.

From the PSNR and SSIM comparison results, firstly we notice that sparse representation based SR methods generally perform better than the interpolation based method (e.g., bicubic), because the over-complete dictionaries can recover high-frequency details of images more accurately.

Image	Measures	Bicubic	ScSR [26]	Zeyde [27]	SCDL [22]	Bilevel [24]	Proposed
Lena	PSNR(dB)	32.7947	34.6874	34.2640	35.1311	35.0680	**35.3308**
	SSIM	0.8872	0.9120	0.9044	0.9140	0.9130	**0.9160**
Mountain	PSNR(dB)	29.6999	31.2343	31.0867	31.4010	31.3757	**31.5459**
	SSIM	0.8430	0.8909	0.8874	0.8937	0.8918	**0.8969**
House	PSNR(dB)	26.3549	27.4334	27.3055	27.6055	27.6115	**27.7919**
	SSIM	0.8048	0.8456	0.8450	0.8510	0.8482	**0.8525**
Lion	PSNR(dB)	30.9312	32.5090	32.4216	32.6028	32.5993	**32.8818**
	SSIM	0.8439	0.8941	0.8926	0.8940	0.8929	**0.8979**
Car	PSNR(dB)	30.5383	32.5275	32.3576	32.5904	32.8914	**33.1157**
	SSIM	0.9138	0.9381	0.9370	0.9396	0.9419	**0.9436**

Table 1. Comparison of factor of 2 magnification super-resolution results.

Image	Measures	Bicubic	ScSR [26]	Zeyde [27]	SCDL [22]	Bilevel [24]	Proposed
Lena	PSNR(dB)	30.0986	31.5125	30.9077	31.5900	31.5808	**31.6818**
	SSIM	0.8019	0.8354	0.8156	0.8347	0.8344	**0.8377**
Mountain	PSNR(dB)	27.0522	28.0436	27.8258	28.0490	28.0606	**28.1259**
	SSIM	0.7000	0.7596	0.7520	0.7607	0.7561	**0.7636**
House	PSNR(dB)	24.4172	25.0136	24.7198	25.0100	25.0277	**25.0592**
	SSIM	0.6881	0.7230	0.7234	0.7236	0.7235	**0.7248**
Lion	PSNR(dB)	28.3921	29.0637	29.0455	29.0483	29.1161	**29.2190**
	SSIM	0.7058	0.7496	0.7498	0.7512	0.7473	**0.7537**
Car	PSNR(dB)	27.4234	28.6083	28.5011	28.4892	28.7231	**28.8557**
	SSIM	0.8259	0.8630	0.8573	0.8635	0.8652	**0.8673**

Table 2. Comparison of factor of 3 magnification super-resolution results.

Next, Zeyde's [27] two-step learned dictionaries have the similar performance as the coupled learned dictionaries (ScSR) [26], while the most recent semi-coupled dictionary learning methods SCDL [22] and Bilevel [24] outperform the coupled dictionary learning algorithm in both PSNR and SSIM. Finally, the proposed BP-JDL method further pushes the limit by providing a flexible and consistent learning model, and is able to provide high-res images with the best recover accuracy.

From the visual comparison results, we also notice that generally sparse representation based SR methods produce sharper image than bicubic interpolation. Next, we notice the improvement of SCDL and Bilevel methods compared to the ScSR method in terms of artifacts on the edges. Among the results of all sparse representation based methods, images produced by the proposed BP-JDL algorithm have the least artifacts, indicating that the proposed method can better restore the high-res images from low-res images.

During the SR reconstruction process, theoretically the more overlap of patches, the better the SR results. SR results of different overlap values are shown in Fig. 2. The results demonstrate the positive relationship between the overlap size and PSNR (SSIM), confirming using maximum overlap (patchsize - 1) can generate the best SR results.

The average factor of 2 SR reconstruction time of ScSR, Zeyde, SCDL, Bilevel and BP-JDL are 217.9s, 1.9s, 1837.8s, 218.7s and 213.5s, respectively. Results were produced on a Dell T3500 with 2.66G CPU and 12GB RAM running Matlab V7.12.0. Among these methods, the Zeyde method is the fastest. Although BP-JDL benefits from using a smaller dictionary compared to ScSR, the extra operation of Eq. 7 consumes extra time. However, BP-JDL is still faster than ScSR, SCDL and Bilevel methods. SCDL is the slowest method because it needs 32 dictionaries (clusters) for each feature space instead of single dictionary, thus consuming much more time than other methods.

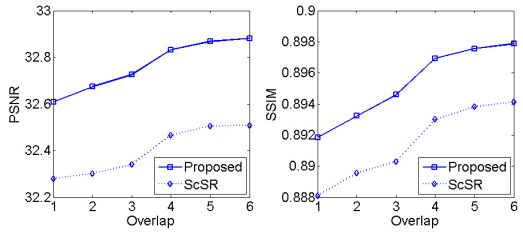

Figure 2. Effect of the overlap parameter on PSNR and SSIM of test image Lion.

5. Conclusion

In this paper, a beta process joint dictionary learning (BP-JDL) method was proposed for solving the dictionary learning problem in coupled feature spaces. The proposed method could have wide applications in the field of signal processing because many problems in this area require the mapping between two feature spaces. We applied this method to solve the single image super resolution (SISR) problem. Four state-of-the-art dictionary learning based SISR methods were compared with BP-JDL in terms of the quality of dictionary generated and the quality of the super-resolution images. The experimental results showed that the BP-JDL method is able to learn dictionaries that fit the coupled feature spaces better than previous methods. The SISR results showed that the images reconstruction using BP-JDL have the best overall quality compared to other four meth-

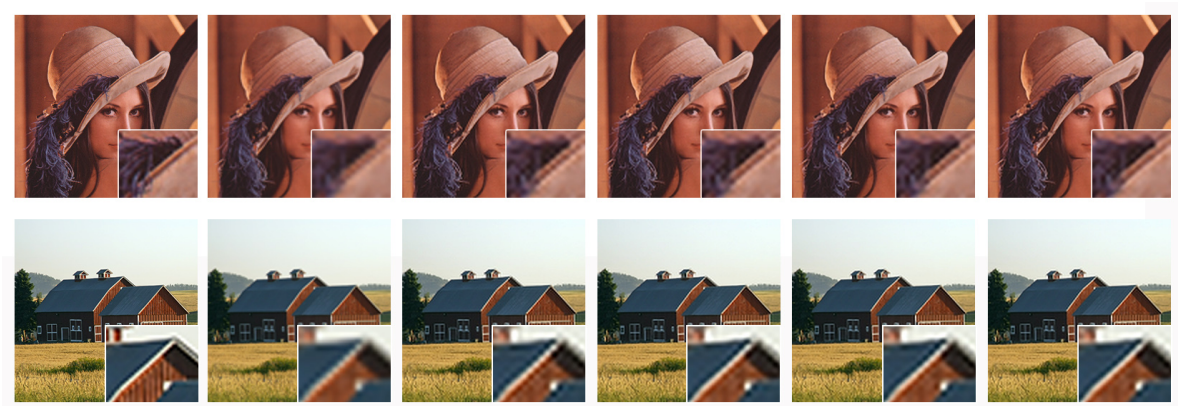

Figure 3. Visual comparison of factor of 2 super-resolution results. The upper row shows the SR results of the image Lena. The lower row shows the SR results of the image House.

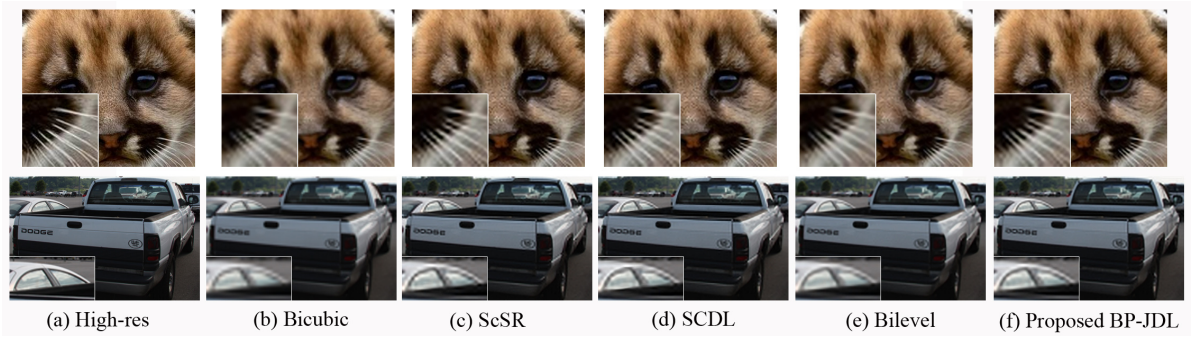

Figure 4. Visual comparison of factor of 3 super-resolution results. The upper row shows the SR results of the image Lion. The lower row shows the SR results of the image Car.

ods. In addition, BP-JDL was able to infer an appropriate dictionary size non-parametrically. In the future, a variational Bayesian inference could be used for the BP-JDL inference, which may have a faster convergence speed than Gibbs sampler.

6. Acknowledgements

The authors would like to thank Jianchao Yang from the University of Illinois at Urbana-Champaign, Mingyuan Zhou and John Paisley from Duke University and Shenlong Wang from Northwestern Polytechnical University for providing code and discussion.

Appendices

A. Gibbs Sampling Inference

The Gibbs sampling update equations for BP-JDL are given below.

- Sample $\mathbf{d}_k^{(x)}$ from $p(\mathbf{d}_k^{(x)}|-) \sim \mathcal{N}(\mu_{\mathbf{d}_k^{(x)}}, \Sigma_{\mathbf{d}_k^{(x)}})$

$$\begin{aligned} \Sigma_{\mathbf{d}_k^{(x)}} &= (P_x \mathbf{I} + \gamma_\epsilon^{(x)} \sum_{i=1}^{N} z_{ik}^2 {s_{ik}^{(x)}}^2)^{-1} \\ \mu_{\mathbf{d}_k^{(x)}} &= \gamma_\epsilon^{(x)} \sum_{i=1}^{N} z_{ik} s_{ik}^{(x)} \mathbf{x}_i^{-k} \end{aligned} \tag{10}$$

where $\mathbf{x}_i^{-k} = \mathbf{x}_i - \mathbf{D}(\mathbf{s}_i^{(x)} \circ \mathbf{z}_i) + \mathbf{d}_k^{(x)}(s_{ik}^{(x)} \circ z_{ik})$.

- Sample z_{ik}

$$\begin{aligned} & p(z_{ik} = 1|-) \\ \propto \quad & \pi_k \exp[-\frac{\gamma_\epsilon^{(x)}}{2}({s_{ik}^{(x)}}^2 {\mathbf{d}_k^{(x)}}^T \mathbf{d}_k^{(x)} - 2 s_{ik}^{(x)} {\mathbf{d}_k^{(x)}}^T \mathbf{x}_i^{-k}) \\ & - \frac{\gamma_\epsilon^{(y)}}{2}({s_{ik}^{(y)}}^2 {\mathbf{d}_k^{(y)}}^T \mathbf{d}_k^{(y)} - 2 s_{ik}^{(y)} {\mathbf{d}_k^{(y)}}^T \mathbf{y}_i^{-k})] \\ & p(z_{ik} = 0|-) = 1 - \pi_k \end{aligned} \tag{11}$$

- Sample $s_{ik}^{(x)}$ from $p(s_{ik}^{(x)}|-) \sim \mathcal{N}(\mu_{s_{ik}^{(x)}}, \Sigma_{s_{ik}^{(x)}})$ where

$$\begin{aligned} \Sigma_{s_{ik}^{(x)}} &= (\gamma_s^{(x)} + \gamma_\epsilon^{(x)} z_{ik}^2 \mathbf{d}_k^{(x)T} \mathbf{d}_k^{(x)})^{-1} \\ \mu_{s_{ik}^{(x)}} &= \gamma_\epsilon^{(x)} \Sigma_{s_{ik}^{(x)}} (z_{ik} \mathbf{d}_k^{(x)T} \mathbf{x}_i^{-k}) \end{aligned} \tag{12}$$

- Sample π_k from $p(\pi_k|-) \sim Beta(\pi_k; a, b)$ where $a = \frac{a_0}{K} + \sum_{i=1}^N z_{ik}$ and $b = \frac{b_0(K-1)}{K} + N - \sum_{i=1}^N z_{ik}$.

- Sample $\gamma_s^{(x)}$ from a Gamma distribution as

$$p(\gamma_s^{(x)}|-) \sim \Gamma(c_0 + \frac{1}{2}KN, d_0 + \frac{1}{2}\sum_{i=1}^N \|\mathbf{s}_i^{(x)T} \mathbf{s}_i^{(x)}\|) \tag{13}$$

- Sample $\gamma_\epsilon^{(x)}$ from a Gamma distribution as

$$p(\gamma_\epsilon^{(x)}|-) \sim \Gamma(e_0 + \frac{1}{2}N, f_0 + \frac{1}{2}\sum_{i=1}^N \|\mathbf{x}_i^{-k}\|^2) \tag{14}$$

The $\mathbf{d}_k^{(y)}$, $s_{ik}^{(y)}$, $\gamma_s^{(y)}$ and $\gamma_\epsilon^{(y)}$ can be sampled in similar way of $\mathbf{d}_k^{(x)}$, $s_{ik}^{(x)}$, $\gamma_s^{(x)}$ and $\gamma_\epsilon^{(x)}$, respectively.

References

[1] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. on Signal Processing*, 54(11), 2006. 1, 2, 3

[2] A. Buades, B. Coll, and J.-M. Morel. A non-local algorithm for image denoising. In *Proc. of CVPR*, volume 2, pages 60 – 65 vol. 2, 2005. 4, 5

[3] D. Capel. Image mosaicing and super-resolution. *Ph.D. Thesis, University of Oxford*, 2001. 4

[4] H. Chang, D.-Y. Yeung, and Y. Xiong. Super-resolution through neighbor embedding. In *Proc. of CVPR*, volume 1, 2004. 2

[5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Trans. on Image Processing*, 16(8):2080 –2095, aug. 2007. 4

[6] T. L. Griffiths and Z. Ghahramani. Infinite latent feature models and the indian buffet process. In *Proc. of NIPS*, 2005. 2

[7] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. Image analogies. In *Proc. of SIGGRAPH*, pages 327–340, 2001. 2

[8] D. Knowles and Z. Ghahramani. Infinite sparse factor analysis and infinite independent components analysis. In *Independent Component Analysis and Signal Separation*, volume 4666. 2007. 3

[9] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *Proc. of NIPS*, 2007. 1, 2, 4

[10] Z. Lei and S. Li. Coupled spectral regression for matching heterogeneous faces. In *Proc. of CVPR*, pages 1123 –1128, 2009. 2

[11] D. Lin and X. Tang. Coupled space learning of image style transformation. In *Proc. of ICCV*, volume 2, pages 1699 – 1706 Vol. 2, 2005. 2

[12] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *Proc. of the ICML*, 2009. 1, 2

[13] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Trans. on Signal Processing*, 41(12):3397 –3415, 1993. 1

[14] J. F. Murray and K. Kreutz-Delgado. Learning sparse overcomplete codes for images. *J. VLSI Signal Process. Syst.*, 46, 2007. 1

[15] B. A. Olshausen and D. J. Fieldt. Natural image statistics and efficient coding. *Network Bristol England*, 7(2), 1996. 1

[16] B. A. Olshausen and D. J. Fieldt. Sparse coding with an overcomplete basis set: a strategy employed by v1. *Vision Research*, 37, 1997. 1

[17] J. Paisley and L. Carin. Nonparametric factor analysis with beta process priors. In *Proc. of ICML*, 2009. 1, 2, 3

[18] P. Rai and H. Daumé III. The infinite hierarchical factor regression model. In *Proc. of NIPS*, 2008. 3

[19] M. Ranzato, C. Poultney, S. Chopra, and Y. LeCun. Efficient learning of sparse representations with an energy-based model. In *Proc. of NIPS*, 2006. 1

[20] J. Sun, N.-N. Zheng, H. Tao, and H.-Y. Shum. Image hallucination with primal sketch priors. In *Proc. of CVPR*, volume 2, 2003. 2

[21] M. E. Tipping. Sparse bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.*, 1, 2001. 1, 2

[22] S. Wang, L. Zhang, Y. Liang, and Q. Pan. Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis. In *Proc. of CVPR*, 2012. 1, 2, 4, 6

[23] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. on Image Processing*, 13(4):600 –612, 2004. 5

[24] J. Yang, Z. Wang, Z. Lin, X. Shu, and T. Huang. Bilevel sparse coding for coupled feature spaces. In *Proc. of CVPR*, 2012. 1, 2, 3, 4, 6

[25] J. Yang, J. Wright, T. Huang, and Y. Ma. Image super-resolution as sparse representation of raw image patches. In *Proc. of CVPR*, 2008. 1, 2, 4

[26] J. Yang, J. Wright, T. Huang, and Y. Ma. Image super-resolution via sparse representation. *IEEE Trans. on Image Processing*, 19(11), 2010. 1, 3, 4, 5, 6

[27] R. Zeyde, M. Elad, and M. Protter. On single image scale-up using sparse-representation. In *Proc. of Internation Conference on Curves and Surfaces*, 2010. 1, 2, 6

[28] M. Zhou, H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, D. Dunson, G. Sapiro, and L. Carin. Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images. *IEEE Trans. on Image Processing*, 21(1):130 –144, 2012. 2, 3, 4