Towards Sparsity and Selectivity: Bayesian Learning of Restricted Boltzmann Machine for Early Visual Features*

Hanchen Xiong, Sandor Szedmak,
Antonio Rodríguez-Sánchez, and Justus Piater

Institute of Computer Science, University of Innsbruck
{hanchen.xiong, sandor.szedmak, antonio.rodriguez-sanchez,
justus.piater}@uibk.ac.at

Abstract. This paper exploits how Bayesian learning of restricted Boltzmann machine (RBM) can discover more biologically-resembled early visual features. The study is mainly motivated by the sparsity and selectivity of visual neurons' activations in V1 area. Most previous work of computational modeling emphasize selectivity and sparsity independently, which neglects the underlying connections between them. In this paper, a prior on parameters is defined to simultaneously enhance these two properties, and a Bayesian learning framework of RBM is introduced to infer the maximum posterior of the parameters. The proposed prior performs as the lateral inhibition between neurons. According to our empirical results, the visual features learned from the proposed Bayesian framework yield better discriminative and generalization capability than the ones learned with maximum likelihood, or other state-of-the-art training strategies.

1 Introduction

Over the past decades, there have been a large volume of literature dedicated to model the statistics of natural images in biologically plausible ways. Especially, the primary visual cortex (V1) has been intensively studied and various computational models were proposed to reproduce its functionalities [10,6,12]. It has been well documented that mainly V1 simple cells perform an early stage processing of the visual input signal from the retina and the lateral geniculate nucleus (LGN). One important property of V1 simple cells is that their receptive fields are selective in terms of locations, orientations and frequencies, which can be modelled as Gabor filters. Another characteristic on V1 simple cells is that their activations are sparse. To be more clear, selectivity means that one neuron

* The authors would like to thank Dr. George Azzopardi for his helpful comments. The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 (Specific Programme Cooperation, Theme 3, Information and Communication Technologies) under grant agreement no. 270273, Xperience.

only strongly responds to a small number types of stimuli while rarely responding to other types. Sparsity means that the population size of activated neurons should be small given a stimulus, i.e. only a tiny fraction of neurons are activated by a stimulus. Since selectivity and sparsity are interpreted as rareness in lifetime and population domain, sometimes they are also called "lifetime sparseness" and "population sparseness" respectively [12]. It has been hypothesized that the selective and sparse responses of visual neurons are due to certain redundancy reduction mechanism, with which the visual cortex is evolved to encode visual information as efficiently as possible [1]. Based on this hypothesis, a sparse coding strategy was proposed to enhance the coding efficiency, and has led to Gabor-like representations [10]. Although sparse coding has shown success in producing receptive fields similar to those of simple cells, yet it was pointed out that selectivity does not have to be correlated with sparseness in practice [12]. Moreover, it was even suspected that sparse activations of simple cells is only an epiphenomenon or side effect of selectivity [2]. (see section 3 for a detailed analysis). Recently, as another stream of feature learning, restricted Boltzmann machines (RBMs) have attracted increasingly more attention thanks to its success in many application domains [7]. However, the capability of RBMs is rather limited when learning receptive fields of V1 simple cells. To make inference and learning easier, there is no connection between hidden units in RBMs. Consequently, given visible data, all hidden units are conditionally independent to each other (see section 2). It can be easily envisioned that when RBMs are trained on natural images, many learned features will be rather distributed, unlocalized and repeated, which is far from the (selective and sparse) nature of the learning task.

Prior work have exploited different strategies to adapt RBMs towards learning selective and/or sparsely activated neurons [8,9,5] on visual input. However, most of them focus only on one property and does not ensure sparsity and selectivity simultaneously in reproduced neurons. Usually, these strategies are to impose certain regularization to bias learning. To overcome this deficiency, in this paper, we propose to encode an inductive bias about the task as prior probability on parameters. Then, the parameter estimation can be done within a consistent Bayesian learning framework, *i.e.* maximum a posterior (MAP). In particular, the prior probability on parameters encourages the diversity of neurons' receptive fields, which performs equivalently to the lateral inhibition between neurons. The MAP learning is achieved via a Markov chain Monte Carlo (MCMC)-based simulated annealing. In addition, due to the fact that the parameter space is high-dimensional and multi-modal, annealing importance sampling (AIS) and parallel tempering are employed in subroutines to avoid local maxima (see section 4). In section 5, we verify our Bayesian learning of RBMs on a benchmark database of natural images, comparing to maximum likelihood learning and other state-of-the-art learning strategies. Our empirical results demonstrate that neurons in our model display better sparsity and selectivity than in others; in addition, the features encoded by our neurons via Bayesian learning show better generalization capabilities than the ones from other learning methods.

2 Restricted Boltzmann Machine

The restricted Boltzmann machine (RBM) is a two-layer, bipartite neural network, it is a "restricted version" of the Boltzmann machine with only interconnections between hidden layers and visible layers. Input data is binary and N_v dimensional, they are fed into N_v units in the visible layer $\mathbf{v}$, N_h units in the hidden layer $\mathbf{h}$ are stochastic binary variables, *i.e.* $\mathbf{v} \in \{0,1\}^{N_v}$, $\mathbf{h} \in \{0,1\}^{N_h}$, the joint probability of $\{\mathbf{v}, \mathbf{h}\}$ is[1]:

$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{\mathbf{Z}} \exp(-E(\mathbf{v}, \mathbf{h})) \qquad E(\mathbf{v}, \mathbf{h}) = -\mathbf{v}^\top \mathbf{W}\mathbf{h} \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{N_v \times N_h}$ is the matrix of symmetry weights, $\mathbf{Z} = \sum_{\mathbf{v},\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h}))$ is the partition function for normalization. Because of the restricted connections in RBMs, hidden units h_j are independent of each other conditioned on the visible data $\mathbf{v}$, and similarly, visible units v_i are conditionally independent of each other given $\mathbf{h}$. Given training data $\mathcal{D} = \{\mathbf{v}^{(l)}\}_{l=1}^{L}$, RBM can be learned by maximizing the average log-likelihood of $\mathcal{D}$:

$$\mathbf{W}^* = \arg\max_{\mathbf{W}} \mathcal{L}(\mathcal{D}) = \arg\max_{\mathbf{W}} \frac{1}{L} \sum_{l=1}^{L} \big(\log \sum_{\mathbf{h}} p(\mathbf{v}^l, \mathbf{h}) \big) \tag{2}$$

based on(1), the gradient of $\mathcal{L}(\mathcal{D})$ is computed as:

$$\nabla \mathcal{L}(\mathcal{D}) = \frac{1}{L} \sum_{l=1}^{L} [\mathbb{E}_{\mathbf{v}^{(l)} \in \mathbf{V}, \mathbf{h} \sim p(\mathbf{h}|\mathbf{v}^{(l)})}(\mathbf{v}^{(l)}\mathbf{h}^\top) - \mathbb{E}_{\mathbf{v},\mathbf{h} \sim p(\mathbf{v},\mathbf{h})}(\mathbf{v}\mathbf{h}^\top)] \tag{3}$$

where $\mathbb{E}_p(\cdot)$ denotes the expected values with respect to p. Obviously, the sampling $\mathbf{v}, \mathbf{h} \sim p(\mathbf{v}, \mathbf{h})$ makes learning practically infeasible because it requires a large number of Markov chain Monte Carlo (MCMC) iterations to reach equilibrium. Fortunately, we can compute an efficient approximation to the exact gradient: contrastive divergence (CD), which works well in practice [7]. By using CD_k, only a small number k steps are run in block Gibbs sampling (usually $k = 1$), and (3) can be approximated as:

$$\nabla \hat{\mathcal{L}}(\mathcal{D}) = \frac{1}{L} \sum_{l=1}^{L} [\mathbf{v}^{(l)} p(\mathbf{h}^{(l)+} | \mathbf{v}^{(l)})^\top - p(\mathbf{v}^{(l)-} | \mathbf{h}^{(l)+}) p(\mathbf{h}^{(l)-} | \mathbf{v}^{(l)-})^\top] \tag{4}$$

3 Bias Learning with Selectivity and Sparsity

Simple cells in V1 area are well known to be selective to locations, orientations and frequencies, and their activations are sparse [10] to visual stimuli. The concepts of selectivity and sparsity are illustrated in Figure 1(a), where each row (red) represent how one neuron selectively respond to different visual stimuli while each column (blue) describes how many neurons are activated by one stimulus. Although selectivity and sparsity are related at their average values,

[1] Bias vectors on visible and hidden units are omitted them for notation simplicity, but we would like to note that we use such biases in our experiments.

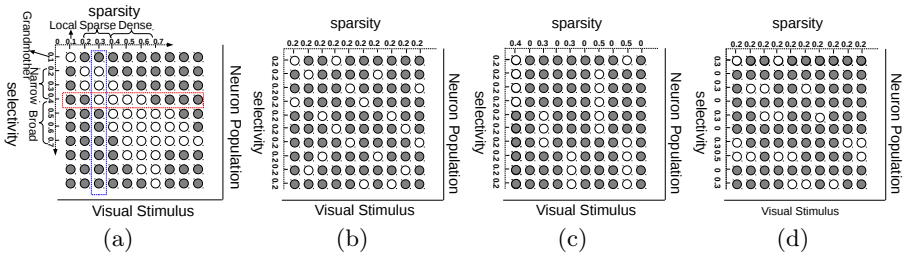


Fig. 1. Understanding sparsity and selectivity: white circles mean activations while gray circles denote inactivations. See text for more description.

they are not necessarily correlated [12]. Selective neurons cannot ensure sparse neuron coding (Figure 1(c)); sparsely activated neurons can also be not narrowly selective (Figure 1(d)).

Sparse group restricted Boltzmann machine (SGRBM) [9] is a RBM trained with the CD algorithm plus a $l1/l2$ norm regularization on the activations of neuron population. Although $l1/l2$ norm regularization can ensure sparsity, yet it can also lead to many "dead" (never respond) and "potential over-tolerant" (always respond) neurons (see Figure 1(d) and section 5). On the other hand, a selectivity-induced regularization was used in [8] by suppressing the average activation probability of each neuron to all training data. One limitation of this strategy, as argued in [5], is that decreasing average activation probabilities can not guarantee selectivity, instead, it will result in many similar neurons with uniformly low activation probabilities to all types of visual stimuli, which prone to be "dead" as well (see section 5). One closely related work to ours is proposed in [5], of which the essence is to tune the activation matrix (Figure 1) towards a target one that is both selective and sparse while maximizing likelihood.

Based on the analysis above, we can see that the motivation of sparsity is to better differentiate neurons while the goal of selectivity is to avoid "over-tolerant" neurons. Assume that there exist N types of visual stimuli and K neurons, and usually $N \gg K$. Obviously, the ideal selectivity rates of neurons is N/K. At the same time, for sparsity, we also want to prevent the existence of any duplicate or similar activations in the neuron population. The best scenario is that there is no overlap among the activations of different neurons (rows in Figure 1), *i.e.* K neurons respond to non-overlapping N/K types of visual stimuli respectively. In the RBM case, the weights W, to some extent, can represent the activation matrix. For so, a natural choice of biasing parameters is to diversify the columns of $\mathbf{W}$ as much as possible. Here we approach diversification by minimizing absolute cosine similarities among columns of $\mathbf{W}$:

$$\arg\min_{\mathbf{W}} \sum_{j=1}^{N_h} \sum_{k \neq j}^{N_h} \left| \frac{\mathbf{W}_{\cdot,j}^{\top} \mathbf{W}_{\cdot,k}}{||\mathbf{W}_{\cdot,j}|| ||\mathbf{W}_{\cdot,k}||} \right| \tag{5}$$

where $\mathbf{W}_{\cdot j}$ denotes the jth column of $\mathbf{W}$. Note that the denominator in (5) is necessary, because eliminating it will generate many "dead" or *principal component*

analysis (PCA)-like neurons. An extreme case is that the activation probabilities of neurons are exclusive to each other. Despite selectivity is not so obvious in (5), it can be imagined that it can be better minimized when $|W_{\cdot,j:\forall j \in N_h}|$ are small. Therefore, sparsity and selectivity are enhanced simultaneously by using diversity-induced bias (5) (Figure 1(b)).

4 Bayesian Learning of Restricted Boltzmann Machines

In contrast to the incremental updates composed of CD approximation and regularization-based gradients [8,9,5], we propose to train RBMs in a consistent Bayesian framework. Based on the discussion in the previous section, we can define the prior probability on parameters $p(\mathbf{W})$ as:

$$p(\mathbf{W}) \propto \exp(-\lambda \cdot \sum_{j=1}^{N_h} \sum_{k \neq j}^{N_h} \left| \frac{\mathbf{W}_{\cdot,j}^{\top} \mathbf{W}_{\cdot,k}}{||\mathbf{W}_{\cdot,j}|| ||\mathbf{W}_{\cdot,k}||} \right|) \tag{6}$$

then the parameters can be estimated via maximum a posterior (MAP):

$$\mathbf{W}^* = \arg\max_{\mathbf{W}} p(\mathbf{W}|\mathcal{D}) = \arg\max_{\mathbf{W}} p(\mathbf{W}) \prod_{l=1}^{L} \sum_{\mathbf{h}} p(\mathbf{v}^{(l)}, \mathbf{h}|\mathbf{W}) \tag{7}$$

Since the derivative of (7) *w.r.t.* $\mathbf{W}$ can not be analytically computed, and (7) in general is not concave, here a Markov chain Monte Carlo (MCMC)-based simulated annealing is employed to find the optimal solution. In the basic Metropolis algorithm, a sample $\mathbf{W}'$ is accepted with probability $\min(1, p(\mathbf{W}'|\mathcal{D})/p(\mathbf{W}|\mathcal{D}))$ where:

$$\frac{p(\mathbf{W}'|\mathcal{D})}{p(\mathbf{W}|\mathcal{D})} = \frac{p(\mathbf{W}')}{p(\mathbf{W})} \frac{p(\mathcal{D}|\mathbf{W}')}{p(\mathcal{D}|\mathbf{W})} = \frac{p(\mathbf{W}')}{p(\mathbf{W})} \frac{\prod_{l=1}^{L} \sum_{\mathbf{h}} p(\mathbf{v}^{(l)}, \mathbf{h}|\mathbf{W}')}{\prod_{l=1}^{L} \sum_{\mathbf{h}} p(\mathbf{v}^{(l)}, \mathbf{h}|\mathbf{W})} \tag{8}$$

Because of the special structure of RBM, the term $\sum_{\mathbf{h}} p(\mathbf{v}, \mathbf{h}|\mathbf{W})$ can be written in a polynomial form as:

$$p(\mathbf{v}|\mathbf{W}) = \sum_{\mathbf{h}} p(\mathbf{v}, \mathbf{h}|\mathbf{W}) = \frac{1}{\mathbf{Z}(\mathbf{W})} \prod_{j=1}^{N_h} \exp(1 + \mathbf{v}^{\top} \mathbf{W}_{\cdot,j}) \tag{9}$$

Consequently, (8) can be further expanded as:

$$\frac{p(\mathbf{W}'|\mathcal{D})}{p(\mathbf{W}|\mathcal{D})} = \frac{p(\mathbf{W}')}{p(\mathbf{W})} \Big(\frac{\mathbf{Z}(\mathbf{W})}{\mathbf{Z}(\mathbf{W}')} \Big)^N \exp \Big\{ \sum_{l=1}^{L} \sum_{j=1}^{N_h} \mathbf{v}^{(l)\top} (\mathbf{W}'_{\cdot,j} - \mathbf{W}_{\cdot,j}) \Big\} \tag{10}$$

Since (10) is invariant to different scales of $\mathbf{W}$, without loss of generality, we constraint $\forall w_{ij} \in \mathbf{W}, w_{i,j} \in [-1, +1]$. One difficulty in computing (10) is the ratio of normalization terms $\frac{\mathbf{Z}(\mathbf{W})}{\mathbf{Z}(\mathbf{W}')}$. Instead of computing it analytically, we use a tractable approximation of it via *annealing importance sampling* (AIS) [11]. Basically, importance sampling can be used for estimating the ratio:

$$\frac{\mathbf{Z}(\mathbf{W})}{\mathbf{Z}(\mathbf{W}')} = \frac{\sum_{\mathbf{v}} p(\mathbf{v}|\mathbf{W})}{\sum_{\mathbf{v}} p(\mathbf{v}|\mathbf{W}')} = \sum_{\mathbf{v}} \frac{p(\mathbf{v}|\mathbf{W})}{p(\mathbf{v}|\mathbf{W}')} \frac{p(\mathbf{v}|\mathbf{W}')}{\sum_{\mathbf{v}} p(\mathbf{v}|\mathbf{W}')} = \mathbb{E}_{p(\mathbf{v}|\mathbf{W}')} \Big(\frac{p(\mathbf{v}|\mathbf{W})}{p(\mathbf{v}|\mathbf{W}')} \Big) \tag{11}$$

However, the estimation will be poor if $\mathbf{W}$ and $\mathbf{W}'$ are not close. By contrast, AIS constructs many intermediary distributions between $p(\mathbf{v}|\mathbf{W}')$ and $p(\mathbf{v}|\mathbf{W})$ as: $p_s(\mathbf{v}) \propto p(\mathbf{v}|\mathbf{W}')^{1-\alpha_s} p(\mathbf{v}|\mathbf{W})^{\alpha_s}$ with $0 < \alpha_0 < \alpha_1 < \cdots < \alpha_s < \cdots < \alpha_S = 1$. Then one AIS run is as follows:

1.Initialize $\mathbf{v}_0^{(m)} \sim p_0(\mathbf{v})$
2.**for** $s = 1 \to S$, sample $\mathbf{v}_s^{(m)}$ give $\mathbf{v}_{s-1}^{(m)}$ with one Gibbs sampling *w.r.t.* $p_s(\mathbf{v})$;
3.$w^{(m)} = \frac{p_1(\mathbf{v}_1^{(m)})}{p_0(\mathbf{v}_1^{(m)})} \frac{p_2(\mathbf{v}_2^{(m)})}{p_1(\mathbf{v}_2^{(m)})} \cdots \frac{p_S(\mathbf{v}_S^{(m)})}{p_{S-1}(\mathbf{v}_S^{(m)})}$.

When M runs of AIS are implemented, the ratio can be estimated as:

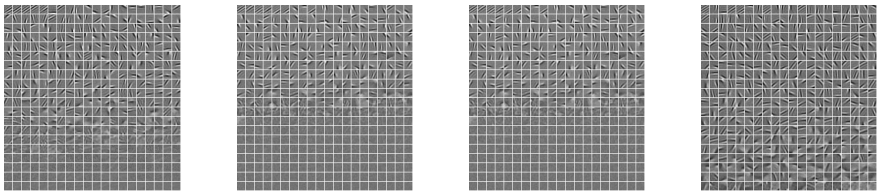
$$\frac{\mathbf{Z}(\mathbf{W})}{\mathbf{Z}(\mathbf{W}')} \approx \frac{1}{M} \sum_{m=1}^{M} w^{(m)} \tag{12}$$

In addition, to avoid being trapped in local maxima, we construct the state transition of a Markov chain as a mixture of a local Metropolis kernel (10) and an independent Metropolis-Hasting kernel. To better explore the sampling space, the uniform distribution $\mathcal{U}$ on $\mathbf{W}$ is set as the Metropolis-Hasting kernel. Therefore, the whole sampling is a weighted combination of local exploitation and global exploration, and here we use the mixture weight $\eta = 0.5$. At iteration n, the invariant distribution which the Markov chain is subject to is $p(\mathbf{W}|\mathcal{D})^{1/T_n}$, where T_n is a decreasing temperature schedule. When $T_n \to 0$, the Markov chain can hardly move and the still state will be used as the maximum. Usually $\mathbf{W}$ is high-dimensional (with large number of neurons and high-dimensional visual input), so the parameter space can be rather complicated, *e.g.* sharp with many isolated modes, and simulated annealing based on one single Markov chain is unreliable. One simple way is to run multiple Markov chains in parallel, and pick the states of one chain which lead to the best result. However, a better strategy is *parallel tempering* [3]. $R+1$ Markov chains are constructed under different initial temperatures $\{p_r(\mathbf{W}|\mathcal{D}) \propto p(\mathbf{W}|\mathcal{D})^{\beta_r}\}_{r=0}^{R}$, $0 \leq \beta_R < \cdots \beta_r < \cdots \beta_0 = 1$, β_0 is referred to the base distribution, and others correspond to more flat distributions smoothed with different temperatures. As the simulated annealing on differently tempered Markov chains progress, the states of neighbouring chains $\mathbf{W}^r$, $\mathbf{W}^{r+1}$ can be swapped with probability:

$$\min(1, \frac{p_r(\mathbf{W}^{r+1}|\mathcal{D}) p_{r+1}(\mathbf{W}^r|\mathcal{D})}{p_{r+1}(\mathbf{W}^{r+1}|\mathcal{D}) p(\mathbf{W}^r|\mathcal{D})}) = \min(1, \exp\{\sum_{l=1}^{L} \sum_{j=1}^{N_h} (\beta_r - \beta_{r+1}) \mathbf{v}^{(l)\top} (\mathbf{W}_{\cdot,j}^{r+1} - \mathbf{W}_{\cdot,j}^{r}\})) \tag{13}$$

5 Experiments

To evaluate the proposed learning strategy, a benchmark database [10] was used for training. 100000 small patches (size 14×14) were extracted from random positions of ten whitened images. A sigmoid function was applied on the pixel intensities to fit their values in the range $[0, 1]$; in addition, the patches with variances smaller than 0.1 were filtered out to accelerate training. For comparison, three additional RBMs were trained by using the CD algorithm, the CD



(a) (b) (c) (d)

Fig. 2. The receptive fields of neurons learned from (a) CD algorithm, (b) sparse CD, (c) selective CD and (d)our Bayesian strategy. See text for more description.

	CD	Sparse CD [9]	Selective CD[8]	Bayesian Learning
# Dead Neurons	**0**	68	88	**0**
Ave. Selectivity	0.4582	0.3552	0.4090	**0.3221**
Ave. Sparsity	0.3749	0.1883	0.1939	**0.1671**
Error rate on MNIST[%]	27.21	17.54	19.32	**14.21**

Fig. 3. Performance of different learning methods.

algorithm with sparse regularization (sparse CD) [9] and the CD algorithm with selectivity regularization (selective CD) [8]. For each RBM, 200 hidden neurons were learned and their receptive fields are presented in Figure 2. We can see that many neurons' receptive fields learned from the CD algorithm (Figure 2(a)) are very vague and unlocalized, compared to which, the neurons' receptive fields learned from sparse CD and selective CD (Figure 2(b) and 2(c)) look "clearer" and "sharper". However, both sparse CD and selective CD led to many useless, "dead" neurons. The neurons obtained from our Bayesian learning strategy display rather diverse receptive fields and there seems no "dead" neuron (Figure 2(d)). We roughly obtained the number of "dead" neurons by counting the number of neurons whose maximal activation probabilities to all training visual stimuli is smaller than 0.1, and the results are in the first row of Figure 3.

Selectivity and sparsity are usually measured using activity ratio [4]. For a neuron, its selectivity is computed across all L input visual stimuli: $selectivity = \left(\sum_{l=1}^{L} r_l/L\right)^2 / \left(\sum_{l=1}^{L} r_l^2/L\right)$ where r_l is the activation rate of the neuron given the lth stimulus. The sparsity of activations by one stimulus is computed across all N_h neurons: $sparsity = \left(\sum_{j=1}^{N_h} r_j/N_h\right)^2 / \left(\sum_{j=1}^{N_h} r_j^2/N_h\right)$. We computed the selectivity and sparsity of 4 RBMs on the MNIST patch dataset[2], which contains digit images. Although natural images and digit images are two absolutely different visual domains, we believe that early features encoded in neurons should be able to successfully adapt from one domain to the other. The results were presented Figure 3. It can be seen that our Bayesian learning method yields bet-

[2] Available on http://yann.lecun.com/exdb/mnist.

ter selectivity and sparsity to other cases. Furthermore, to check the practical effectiveness of learned neurons, we use them as basis filters on the digit images for a multi-classification task. Given a digit image, the activations of hidden neurons are computed as input of a softmax function, and its corresponding label is output. The testing results with four sets of neurons are presented in the bottom part of Figure 3. We can see that the features from Bayesian learning yield lower average test error than others, which suggests superior discriminative and generalization capability.

6 Conclusion

A Bayesian learning framework for RBM was put forward based on many state-of-the-art approximation techniques. To mimic V1 simple cells, a diversity-induced prior was introduced on RBMs' parameters, and maximum a posterior learning yields better results than other learning strategies. In particular, the features encoded in learned neurons display nice discriminative and generalization property for domain adaption. As a possible future work direction, we are studying more sophisticated priors to approach other simple neurons' properties.

References

1. Barlow, H.B.: Unsupervised learning. Neural Computation 1, 295–311 (1989)
2. Berkes, P., White, B., Fiser, J.: No evidence for active sparsification in the visual cortex. In: NIPS. pp. 108–116 (2009)
3. Earl, D.J., Deem, M.W.: Parallel tempering: Theory, applications, and new perspectives. Phys. Chem. Chem. Phys. 7, 3910–3916 (2005)
4. Franco, L., Rolls, E.T., Aggelopoulos, N.C., Jerez, J.M.: Neuronal selectivity, population sparseness, and ergodicity in the inferior temporal visual cortex. Biological Cybernetics 96(6), 547–560 (2007)
5. Goh, H., Thome, N., Cord, M.: Biasing restricted Boltzmann machines to manipulate latent selectivity and sparsity. In: NIPS Workshop on Deep Learning and Unsupervised Feature Learning (2010)
6. van Hateren, J.H., van der Schaaf, A.: Independent component filters of natural images compared with simple cells in primary visual cortex. Proceedings of the Royal Society B: Biological Sciences 265(1394), 359C366 (1998), pMC1688904
7. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science 313(5786), 504–507 (Jul 2006)
8. Lee, H., Ekanadham, C., Ng, A.Y.: Sparse deep belief net model for visual area v2. In: NIPS (2007)
9. Luo, H., Shen, R., Niu, C., Ullrich, C.: Sparse group restricted boltzmann machines. In: AAAI (2011)
10. Olshausen, B., Field, D.: Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature 381, 607–609 (1996)
11. Salakhutdinov, R., Murray, I.: On the quantitative analysis of deep belief networks. In: ICML (2008)
12. Willmore, B., Tolhurst, D.: Characterising the sparseness of neural codes. Network:Comput.Neural Syst. 12, 255–270 (2001)