

LEARNING INVARIANT COLOR FEATURES WITH SPARSE TOPOGRAPHIC RESTRICTED BOLTZMANN MACHINES

Hanlin Goh*, Łukasz Kuśmierz†, Joo-Hwee Lim‡

Nicolas Thome, Matthieu Cord

Institute for Infocomm Research
A*STAR, Singapore

Laboratoire d'Informatique de Paris 6
UPMC - Sorbonne Universités, Paris, France

ABSTRACT

Our objective is to learn invariant color features directly from data via unsupervised learning. In this paper, we introduce a method to regularize restricted Boltzmann machines during training to obtain features that are sparse and topographically organized. Upon analysis, the features learned are Gabor-like and demonstrate a coding of orientation, spatial position, frequency and color that vary smoothly with the topography of the feature map. There is also differentiation between monochrome and color filters, with some exhibiting color-opponent properties. We also found that the learned representation is more invariant to affine image transformations and changes in illumination color.

Index Terms— Unsupervised feature learning, invariant features, sparse coding, topographic coding, color features

1. INTRODUCTION

There is a recent bloom in the field of learning deep architectures [1], whereby the input data is represented by a hierarchical network of several layers. A popular deep architecture is the deep belief net (DBN) [2], which stacks restricted Boltzmann machines (RBMs) in a greedy layer-by-layer manner. Each RBM is trained via unsupervised learning and the entire network is subsequently fine-tuned via supervised learning.

In traditional RBMs, features are learned by approximating the maximum likelihood of the data distribution. Sparsity may be used as a regularization criterion to increase feature differentiation and discriminative power [3, 4, 5]. However, the representations are not invariant to input transformations. We propose that if there is structured similarity between the features, then representations will smoothly vary with respect to the transformations and invariance can be achieved.

In this paper, we present a method to regularize RBM learning to achieve with sparseness and topographical orga-

nization. The features learned exhibit invariance to affine image transformations and illumination color, while maintaining differentiation when the transformations are significant.

2. RESTRICTED BOLTZMANN MACHINES

An RBM is a bipartite neural network that represents input data with a layer of hidden units via symmetric weights. An input example k is represented by visible units $v_i^{(k)}$ and the latent representation by hidden units $h_j^{(k)}$. By fixing one layer, activation probabilities of the other layer can be computed:

$$\Pr(h_j^{(k)} | \mathbf{v}^{(k)}) = \text{sigmoid} \left(\sum_{i=1}^I v_i w_{ij} + b_j \right), \quad (1)$$

$$\Pr(v_i^{(k)} | \mathbf{h}^{(k)}) = \text{sigmoid} \left(\sum_{j=1}^J h_j w_{ij} + c_i \right), \quad (2)$$

where $\text{sigmoid}(\cdot)$ is the logistic function and b_j and c_i are biases contributing to hidden and visible units.

The parameters of the RBM are learned via contrastive divergence [6], whereby the maximum likelihood of the data is approximated. Given a training set of K examples, the visible states v_i^+ and hidden states h_j^+ are sampled from the data distribution, while v_i^- and h_j^- are reconstructed states. The parameter update equations are

$$\Delta w_{ij} = \varepsilon (\langle v_i^+ h_j^+ \rangle - \langle v_i^- h_j^- \rangle), \quad (3)$$

$$\Delta b_j = \varepsilon (\langle h_j^+ \rangle - \langle h_j^- \rangle), \quad (4)$$

$$\Delta c_i = \varepsilon (\langle v_i^+ \rangle - \langle v_i^- \rangle), \quad (5)$$

where ε is a learning rate and $\langle \cdot \rangle$ averages over K samples.

Early work of regularizing RBMs focused on achieving sparse representations [3, 4]. Building on those, Goh et al. [5] formulated a general method to regularize RBMs with more precision. These regularizers can be designed based on any inductive principle, such as sparsity and selectivity as originally demonstrated. The new update rules for \mathbf{W} and \mathbf{b} are:

$$\Delta w_{ij} = \varepsilon (\langle v_i^+ s_j \rangle - \langle v_i^- h_j^- \rangle), \quad (6)$$

$$\Delta b_j = \varepsilon (\langle s_j \rangle - \langle h_j^- \rangle), \quad (7)$$

*Hanlin Goh is also with the Laboratoire d'Informatique de Paris 6, UPMC - Sorbonne Universités, Paris, France and the Image & Pervasive Access Lab, CNRS, Singapore-France. Email: hlgoh@i2r.a-star.edu.sg

†Łukasz Kuśmierz is now with AGH University of Science and Technology, Kraków, Poland.

‡Joo-Hwee Lim is also with the Image & Pervasive Access Lab, CNRS UMI 2955, Singapore-France.

where

$$s_j^{(k)} = \phi p_j^{(k)} + (1 - \phi) h_j^{(k)+} \quad (8)$$

and ϕ is a hyperparameter that interpolates the observed hidden activation probabilities \mathbf{H} with the desired activation probabilities \mathbf{P} . The resulting features learned will take on the representational properties defined by the regularizers \mathbf{P} .

3. SPARSE TOPOGRAPHIC REGULARIZATION

The regularization method drives the learning of features to be dependent on both the data distribution \mathbf{V}^+ and the regularizer \mathbf{P} . We adapt an two-layered scheme [7, 8] to regularize the RBM (Fig. 1). From hidden representation \mathbf{H}^+ , we compute a new set of activations $\hat{\mathbf{H}}$ based on fixed topographically pooled weights (see Section 3.1). Subsequently, sparsity is induced in both the temporal and spatial domains to obtain \mathbf{P} (see Section 3.2). \mathbf{P} can then be used to regularize updating of parameters for the RBM.

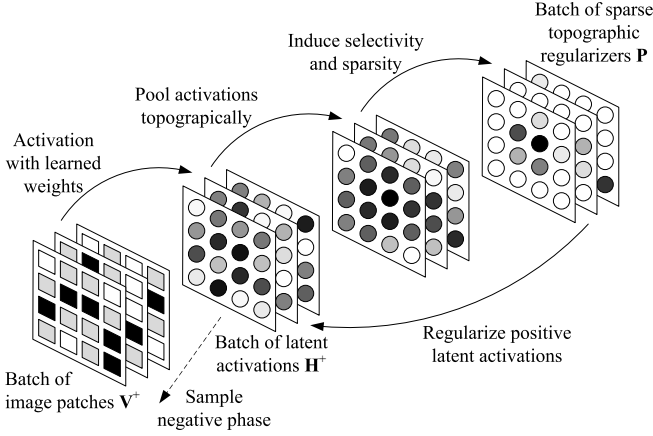


Fig. 1. The framework for inducing both sparseness and topographical organization. From a batch of pixel inputs \mathbf{V}^+ , the latent units are activated \mathbf{H}^+ via learned weights. The activations are then topographically pooled based on the locality of h_j in the feature map via fixed weights. Subsequently, population and lifetime sparseness are induced to obtain \mathbf{P} . Finally, \mathbf{P} is used to regularize the learning of the parameters.

3.1. Inducing Topographic Organization

As shown in Fig. 1, we induce topographical structure in the feature map by introducing a dependence between the hidden units via a new layer $\hat{\mathbf{H}}$, where each $\hat{h}_j^{(k)}$ pools activations from the neighborhood of $h_j^{(k)+}$. Each unit in \mathbf{H}^+ activates units in $\hat{\mathbf{H}}$ depending on the relative locality of the units, which has the same effect as filtering the feature map. The modified activations $\hat{\mathbf{H}}$ are computed as

$$\hat{h}_j^{(k)} = \sum_{m=1}^M h_m^{(k)+} \omega(j, m) \quad (9)$$

where the fixed topographic pooling weights $\omega(\cdot, \cdot)$ are functions of the topographic distance between two units. A Gaussian kernel with wrap around was used for this paper.

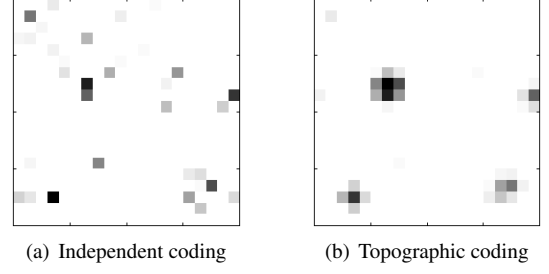


Fig. 2. Comparing activations of independent coding (a) and topographic coding (b). Each pixel shows the activation of a unit in the feature map, where darker color denotes a higher activation. When topographic organization is induced, the activations are spatially grouped within the feature map.

3.2. Inducing Lifetime and Population Sparseness

After inducing topographical organization, we sparsify the activation probabilities to obtain the final modified activation probabilities \mathbf{P} . We follow the method by Goh et al. [5] to induce lifetime and population sparseness in our representations. This is done by the following sequence of data transformations:

$$\hat{p}_j^{(k)} = \left(\text{rank} \left(\hat{h}_j^{(k)}, \hat{\mathbf{h}}^{(k)} \right) \right)^{(1/\mu)-1}, \quad (10)$$

$$p_j^{(k)} = \left(\text{rank} \left(\hat{p}_j^{(k)}, \hat{\mathbf{p}}_j \right) \right)^{(1/\mu)-1}, \quad (11)$$

where μ denotes the target mean of the latent activations and $\text{rank}(x_n, \mathbf{X})$ is the normalized rank of element x_n in vector \mathbf{x} such that the highest x_n is assigned a value of 1, the smallest the value of 0 and all others are uniformly distributed between 0 and 1 depending on their rank in the vector. The $\text{rank}(\cdot)$ function has the same effect as histogram equalization.

4. EXPERIMENTAL RESULTS

We trained an RBM with sparse topographic regularization using 100,000 natural image patches. The patches were of size 10×10 and taken from the McGill Calibrated Colour Image Database [9] (Fig. 3(a)). The RBM consists of 300 visible units and 400 hidden units, which are structured in a two-dimensional 20×20 feature map.

The resulting feature map consists of Gabor-like filters with varying spatial frequency, position, orientation and color (Fig. 4(b)). The appearance of filters vary smoothly across the feature map. This was also demonstrated in other related models [10, 11, 8]. Additionally, our feature map models color information. As a comparison, another RBM was also trained with sparse but independent regularization (Fig. 4(a)).

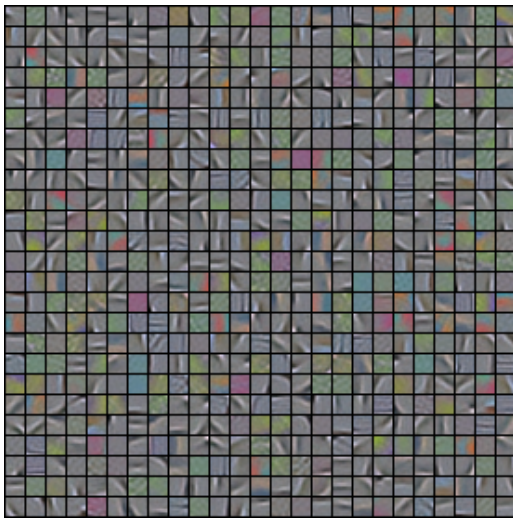


(a) The McGill calibrated color image database has 9 categories – flowers, animals, foliage, textures, fruits, landscapes, winter, man made and shadows.

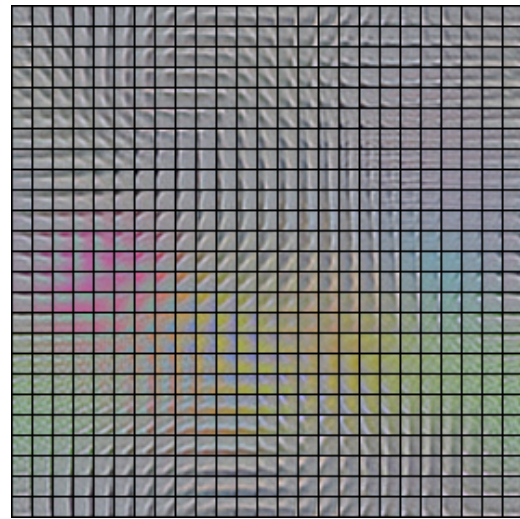


(b) The subset of the Amsterdam library of object images used have been photographed under different illumination color.

Fig. 3. Sample images from data sets used.

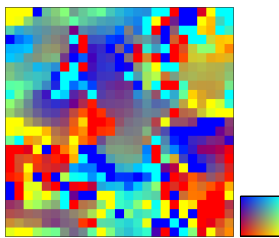


(a) Sparse independent feature map

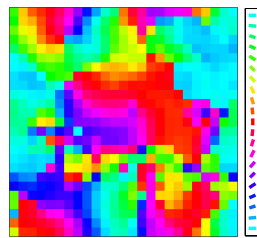


(b) Sparse topographic feature map

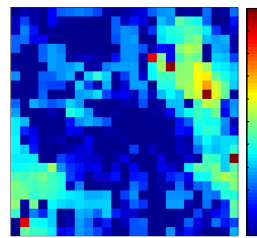
Fig. 4. Feature maps learned with (a) sparse but independent regularizers and (b) sparse topographic regularizers.



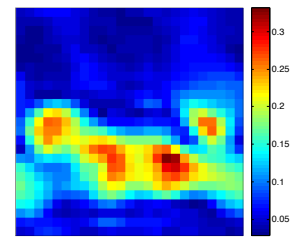
(a) Spatial position



(b) Orientation



(c) Spatial frequency



(d) Color (mean saturation)

Fig. 5. Appearance of filters vary smoothly across the feature map when broken down to their component properties of (a) spatial position, (b) orientation, (c) spatial frequency and (d) color.

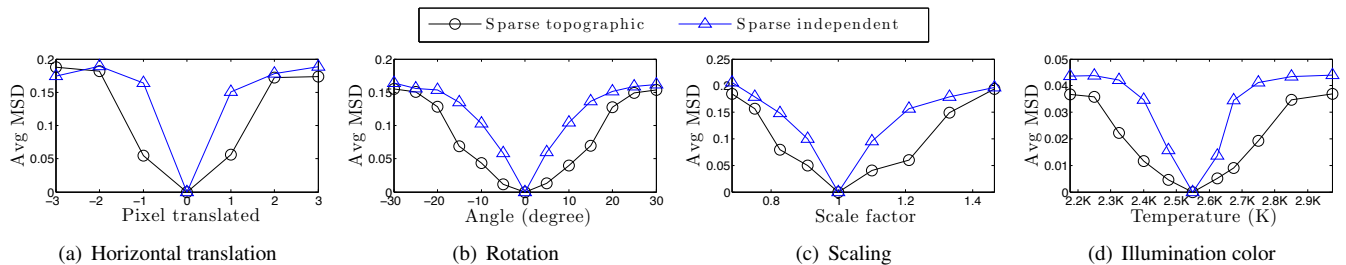


Fig. 6. Comparing the invariance of between sparse topographic features and sparse independent features for (a) translation, (b) rotation, (c) scaling and (d) varying illumination color.

4.1. Analysis of Topographic Feature Maps

We fitted Gabor functions to the filters of the feature map to analyze their orientations, spatial positions and spatial frequencies (Fig 5). At the level of these visual components, one clearly observes a smooth topographical organization of filters in the feature map. There is also a presence of several pinwheel structures in the orientation map (Fig. 5(b)).

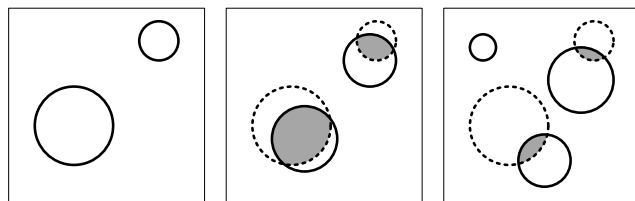
From Fig. 5(d), we can see that the filters are clustered by color. We also observe that some exhibit color-opponency, while others encode single colors. The predominant opponent pairs based-on are red-green, yellow-blue and black-white. From Fig. 5(b), we observe that each color-opponent pair has filters with different orientation coding. The red-green and yellow-blue opponent pairs tend code for lower frequency. We also note the existence of some single-colored filters that are mostly green, cyan and violet, with high frequency textured appearances.

4.2. Evaluating the Invariance of Features

We evaluate the features learned based on their invariance to affine transformations (translation, rotation, scaling) and changes to illumination color. For affine transformations, new patches were sampled from the McGill database [9] (Fig. 3(a)), while patches were drawn from the ALOI data set [12] (Fig. 3(b)) for the illumination color task. For each evaluation task, we extracted 500 evaluation batches. Each evaluation batch consists a set of patches sampled via transformations for the given task.

An evaluation batch for rotation consists of 13 patches sampled about a fixed point with rotation ranging from -30 to 30 degrees at 5 degree intervals. The samples for translation was drawn via horizontal translations from -3 to 3 pixels. For scaling, a progressive scaling factor of $1.1\times$ was used to up-sample and down-sample a patch. To produce patches with varying illumination color, we sampled a set of images produced by different illumination from the same coordinate.

For every input patch, the output signature of hidden unit activations were recorded. To quantitatively measure invariance, we took the mean squared difference (MSD) between the signatures of the transformed input and that of the untransformed input. The MSD was then averaged across the samples and plotted in Fig. 6.



(a) Original signature (b) Slightly transformed (c) Highly transformed

Fig. 7. Variations of signatures in the sparse topographic feature map under different amounts of transformation.

In every evaluation task, when the transformation is low, topographic features are more invariant than independent ones. There is little difference between the two feature types under large transformations. The signature of a slightly transformed input is highly similar the original signature. As the amount of transformation increases, the signature gradually shifts and invariance reduces (Fig. 7). Hence, the features retain the necessary differentiation for recognition tasks.

5. CONCLUSIONS

We introduced a method to regularize the learning of RBMs by encouraging activations to be topographic organized. The appearance of the resulting color features vary smoothly across the feature map. The representations exhibit invariance to affine transformations and changes in illumination color.

6. REFERENCES

- [1] Y. Benjio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, 2009.
- [2] G. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief networks," *Neural Comput.*, 2006.
- [3] H. Lee, C. Ekanadham, and A. Ng, "Sparse deep belief net model for visual area V2," in *NIPS*. 2008.
- [4] V. Nair and G. Hinton, "3D object recognition with deep belief nets," in *NIPS*. 2009.
- [5] H. Goh, M. Cord, and N. Thome, "Biasing restricted Boltzmann machines to manipulate latent selectivity and sparsity," in *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2010.
- [6] G. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, 2002.
- [7] A. Hyvärinen and P. Hoyer, "A two-layer sparse coding model learns simple and complex cell receptive fields and topography from natural images," *Vision Res.*, 2001.
- [8] K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. LeCun, "Learning invariant features through topographic filter maps," in *CVPR*, 2009.
- [9] A. Olmos and F. Kingdom, "McGill calibrated colour image database," 2004, <http://tabby.vision.mcgill.ca>.
- [10] M. Welling, S. Osindero, and G. Hinton, "Learning sparse topographic representations with products of student-t distributions," in *NIPS*. 2003.
- [11] A. Hyvärinen, P. Hoyer, and M. Inki, "Topographic independent component analysis," *Neural Comput.*, 2001.
- [12] J. Geusebroek, G. Burghouts, and A. Smeulders, "The Amsterdam library of object images," *Intl. J. Comput. Vision*, 2005.