Building Efficient Deep Hebbian Networks for Image Classification Tasks

Yanis Bahroun[1], Eugénie Hunsicker[2], and Andrea Soltoggio[1]

[1] Department of Computer Science, Loughborough University
[2] Department of Mathematics, Loughborough University
Loughborough, Leicestershire, United Kingdom
{y.bahroun,e.hunsicker,a.soltoggio}@lboro.ac.uk

To appear in: Proceedings of the 26th International Conference on Artificial Neural Networks (ICANN 2017), Alghero, Italy, 11-15 September, 2017.

Abstract. Multi-layer models of sparse coding (deep dictionary learning) and dimensionality reduction (PCANet) have shown promise as unsupervised learning models for image classification tasks. However, the pure implementations of these models have limited generalisation capabilities and high computational cost. This work introduces the Deep Hebbian Network (DHN), which combines the advantages of sparse coding, dimensionality reduction, and convolutional neural networks for learning features from images. Unlike in other deep neural networks, in this model, both the learning rules and neural architectures are derived from cost-function minimizations. Moreover, the DHN model can be trained online due to its Hebbian components. Different configurations of the DHN have been tested on scene and image classification tasks. Experiments show that the DHN model can automatically discover highly discriminative features directly from image pixels without using any data augmentation or semi-labeling.

Keywords: Sparse coding, Dimensionality reduction, Hebbian/anti-Hebbian learning, MultiDimensional Scaling, Biologically plausible learning rules.

1 Introduction

When applied to supervised learning tasks, deep neural networks trained using backpropagation dominate the field of machine learning in terms of performances on benchmarks. However, such networks often under-perform standard techniques when the number of labelled data available is relatively small. Unsupervised learning, on the contrary, enables the development of algorithms able to adapt to a variety of different unlabeled data sets. For unsupervised learning, a variety of algorithms and principles exist, one of which is the Hebbian principle, stating that in human learning, the connections between two neurons are strengthened when simultaneously activated. Despite the apparent vagueness of this principle, the authors of [20] argue in their work that if rigorously expressed, this principle could be the key to major advances in machine learning. We explicitly express here two important aspects of Hebbian learning that will be used in this work: 1) to be Hebbian, a learning rule should employ only the local information contained in the activities of pre-synaptic and post-synaptic neurons; 2) such learning rules should depend only on the correlation between the activities of these neurons. These two properties of the Hebbian principle are also part of the more general concept of local learning presented in [2].

The work presented here focuses on two unsupervised learning methods, namely, sparse coding and dimensionality reduction. In addition to being powerful statistical learning models, those methods also proved successful at modelling biological signal processing [12, 9]. In this work we have made use of a novel approach [16, 15] that implements both sparse coding and dimensionality reduction by means of a unique principle called *similarity matching*. The minimization of the cost-functions associated, based on Classical Multidimensional Scaling (CMDS) [8], led to trainable neural networks using Hebbian/anti-Hebbian rules.

The work presented here is motivated by two main goals. The first goal is to implement a network for online learning using only feed-forward and lateral connections. The second is to demonstrate that the proposed architecture successfully combines Convolutional Neural Networks (CNN) structure, PCANet [5] and deep sparse coding. In particular, the intent of this work was not to outperform neural networks trained on back-propagation but to evaluate a novel bio-inspired online unsupervised model performing feature extraction for image classification. To achieve these two goals, this study introduces a new type of network called Deep Hebbian Network (DHN) that combines, within one architecture, stages of overcomplete sparse coding and dimensionality reduction based on the *similarity matching* principle. The performance of the DHN is evaluated on indoor scene classification (MIT-67) and image classification (CIFAR-10) tasks.

2 Similarity Matching: a Unifying Framework for Building Efficient Deep Hebbian Networks

The rules implemented in the proposed model derive from adaptations of CMDS. CMDS generates a set of coordinates in a different Euclidean space where the solution is an optimal embedding minimizing the changes to the distances between data points [8]. The formulation of CMDS is given as follows: for a set of inputs $x^t \in \mathbb{R}^n$ for $t \in \{1, \ldots, T\}$, the concatenation of the inputs defines an input matrix $X \in \mathbb{R}^{n \times T}$. The output matrix Y of embeddings is an element of $\mathbb{R}^{m \times T}$. The objective function of CMDS is:

$$Y^* = \underset{Y \in \mathcal{C}}{\arg\min} \, \|X'X - Y'Y\|_F^2 \quad . \tag{1}$$

where F is the Frobenius norm, $X'X$ is the Gram matrix of the inputs, which combines the information of similarity and norm of the vectors, and the space $\mathcal{C}$ encodes the constraints, which depend on the problem to solve. Classically, this method has been used to accomplish dimensionality reduction, and $m < n$. However, it can be adapted to achieve sparse coding. This work focuses on two online versions of CMDS, which leads to non-trivial neural implementations and Hebbian learning rules.

2.1 Hebbian/anti-Hebbian Learning for Similarity Matching

To achieve dimensionality reduction and sparse coding, two different sets of constraints are considered for building the DHN. First, let us assume that the outputs are constrained to be non-negative and of dimension greater than the input dimension, and

$(m > n)$ i.e. $\mathcal{C} = \{Y \in \mathbb{R}_{+}^{m\times T} | m > n\}$. Such constraints correspond to a sparse coding model [16], which optimal solution will be noted Y^{*}_{SC}. Second, if the input dimension is greater than the output dimension, $(n > m)$ and $\mathcal{C} = \{Y \in \mathbb{R}^{m\times T} | m < n\}$, it corresponds to a dimensionality reduction model [15], which optimal solution will be noted Y^{*}_{DR}. In particular, these two optimization problems can be expressed as:

$$Y^{*}_{SC} = \underset{Y\in\mathbb{R}_{+}^{m\times T},\ m>n}{\arg\min} \|X'X - Y'Y\|_F^2 \ , \ Y^{*}_{DR} = \underset{Y\in\mathbb{R}^{m\times T},\ m<n}{\arg\min} \|X'X - Y'Y\|_F^2 \ . \tag{2}$$

Online learning versions of the problems in Eq.2 are expressed as:

$$(y^{T}_{SC})^{*} = \underset{y^T\in\mathbb{R}_{+}^{m},\ m>n}{\arg\min} \|X'X - Y'Y\|_F^2 \ , \ (y^{T}_{DR})^{*} = \underset{y^T\in\mathbb{R}^{m},\ m<n}{\arg\min} \|X'X - Y'Y\|_F^2 , \tag{3}$$

where the inputs are considered as a sequence. When a new element, x^T, is presented to the model, an output, y^T, is generated while keeping the previous y^ts unchanged. The components of the solutions of Eq.3 found in [16, 15] using coordinate descent are:

$$(y^{T}_{i,SC})^{*} = \max\left(W_i^T x^T - M_i^T y^T, 0 \right) \quad , \quad (y^{T}_{i,DR})^{*} = W_i^T x^T - M_i^T y^T \tag{4}$$

$$\text{with} \quad W_{ij}^T = \frac{\sum\limits_{t=1}^{T-1} y_i^t x_j^t}{\sum\limits_{t=1}^{T-1} (y_i^t)^2} \quad ; \quad M_{ij}^T = \frac{\sum\limits_{t=1}^{T-1} y_i^t y_j^t}{\sum\limits_{t=1}^{T-1} (y_i^t)^2} \mathbf{1}_{i\neq j} \quad \forall i \in \{1, \ldots, m\}. \tag{5}$$

W^T and M^T can be found using recursive formulations:

$$W_{ij}^T = W_{ij}^{T-1} + \left(y_i^{T-1}(x_j^{T-1} - W_{ij}^{T-1} y_i^{T-1}) \Big/ \hat{Y}_i^T \right) \tag{6}$$

$$M_{ij\neq i}^T = M_{ij}^{T-1} + \left(y_i^{T-1}(y_j^{T-1} - M_{ij}^{T-1} y_i^{T-1}) \Big/ \hat{Y}_i^T \right) \tag{7}$$

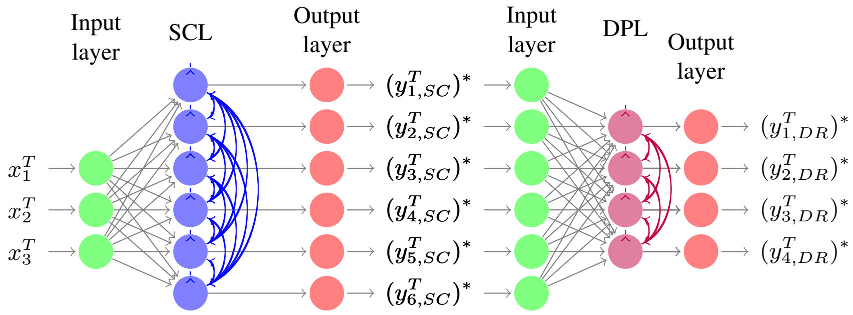
$$\hat{Y}_i^T = \hat{Y}_i^{T-1} + (y_i^{T-1})^2 \quad . \tag{8}$$

The matrices W^T and M^T are sequentially updated using only the relationship between x^{T-1} and y^{T-1}, which are analogous to pre and post-synaptic activities, thus satisfying the Hebbian principle. The learning dynamic of M^T is called anti-Hebbian since the connections between neurons are reduced when activated simultaneously. In both cases the weight matrices W^T and M^T can be interpreted respectively as feed-forward synaptic connections and lateral synaptic inhibitory connections (Fig.1). The main difference between the two models is in the use of a rectified linear unit (ReLU) on the sparse coding problem.

3 Deep Hebbian Network

A DHN is defined here as a combination of three basic layers: a Sparse Coding Layer (SCL), a Depth Pooling Layer (DPL), and a Spatial Pooling Layer (SPL). The different layers can be stacked in various manners to construct a variety of DHNs. Fig. 2 shows a graphical representation of such a network with 2 layers.

Fig. 1: Network implementing successively sparse coding and dimensionality reduction



3.1 Feature Extraction by Sparse Coding: Simple Cell Inspiration

The SCL performs the encoding of local patches using competitive learning modeled by lateral synaptic inhibitions. The choice of layer for extracting features is inspired by the simple cells of the visual cortex V1. It has been proved that overcomplete sparse coding reproduces important tuning properties of those cells [12].

As part of the evaluation of the DHN it is important to assess its performance as a function of the number of SCLs and of the number of neurons in each of those layers. As in an earlier work [1], the sparse coding layers considered perform overcomplete representations of the input data with more output neurons than input neurons, ($m > n$), as expressed in Eq. 3. Overcompleteness in the SCL was chosen because it may allow for more flexibility in matching the output structure to the inputs [12].

3.2 Dimensionality Reduction and Pooling: Complex Cells Inspiration

The model in [1] suffered from the fact that the number of neurons exponentially increases with the number of layers. A key idea in the proposed DHN architecture is to overcome this problem by using dimensionality reduction techniques to reduce the input sizes of the successive SCLs.

Depth Pooling: The DPL performs an online low-dimensional embedding of the data using the *similarity matching* principle. The DPL reduces the number of feature maps before feeding the following SCL.

The introduction of the DPL is inspired by the work of [9], which showed that visual spatial pooling can be learned by Principal Components analysis (PCA) based techniques, reproducing the tuning properties of V1 complex cells. A similar idea, in a supervised learning setup, can be found in the inception layer proposed by [20], which includes a dimensionality reduction stage. Other less bio-inspired dimensionality reduction models, e.g. autoencoders [21], can also be used.

Spatial Pooling: A standard spatial pooling technique is used in this model to reduce the width and height of the feature maps produced after convolution. The max-pooling operation is used after SCL, and no spatial pooling is performed after DPL.

4 Experiments and Results

The results presented here measure the performance of the DHN on classification tasks. A multi-class Support Vector Machine (SVM) [7] classifies the pictures using output vectors obtained by a simple pooling of the feature vectors, Y^*_{SC}, obtained for the input images from the trained network. In particular, given an input image, each neuron in the SCLs produces a new image, called a feature map, which is pooled in quadrants to form 4 terms of the input vector for the SVM as shown in Fig. 2. The linear SVM has been widely used when evaluating the efficiency of unsupervised learning model, on the benchmarks presented below. Although the use of nonlinear classifiers could increase the accuracy, such increase could not be attributed to the efficiency of the DHN.

4.1 Datasets and Preprocessing

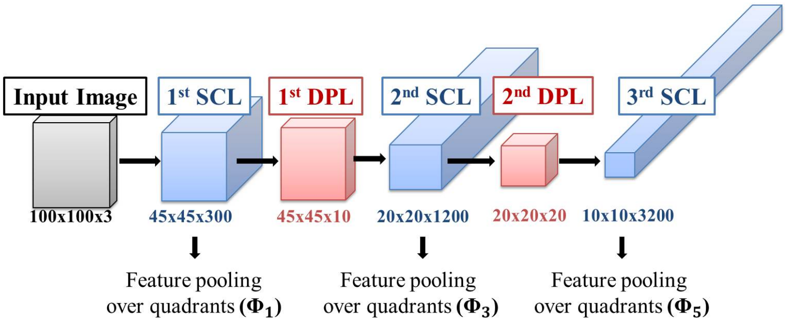
Two datasets are used for evaluating the performance of the features learned by the DHN. The first dataset was the standard benchmark used for indoor scene recognition, the MIT Scene Indoor 67 (MIT-67) [18], which contains 67 indoor categories, with a total of 15620 images. In the following, only 80 images from each class were used for training, and 20 for testing. The second dataset was the CIFAR-10 [10], which contains 50,000 training and 10,000 test images of 32x32 color images of 10 different classes.

Prior to feeding the DHN, basic preprocessing is performed on the inputs, namely brightness and contrast normalization, and whitening. Although online versions of such techniques [15] exist, offline preprocessing is performed in this study to enable a fairer comparison to other unsupervised learning models. A study on the influence of the whitening on the performance of single-layer Hebbian networks is proposed in [1]. The images contained in the MIT-67 are of different resolutions. In order to train and test the DHN on a consistent set of images, the images of the MIT-67 were resized to 100x100x3, size of the smallest image on the dataset.

4.2 Evaluation of DHN

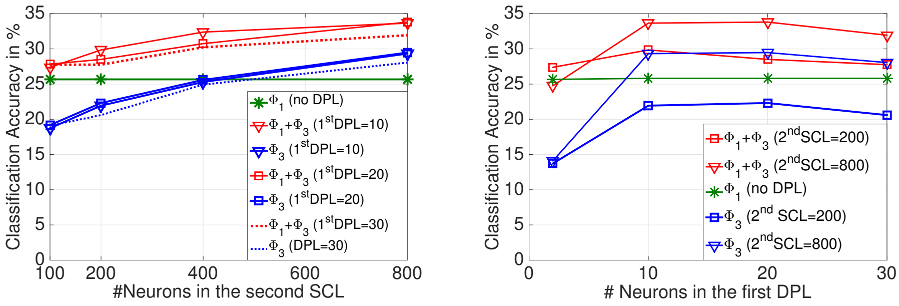
For both datasets, tests were performed on DHNs with up to 5 layers as indicated in the schema in Fig.2.

Fig. 2: DHN with convolutional architecture trained on the MIT-67 dataset.


Input Image
1st SCL
1st DPL
2nd SCL
2nd DPL
3rd SCL
100x100x3
45x45x300
45x45x10
20x20x1200
20x20x20
10x10x3200
Feature pooling over quadrants (Φ_1)
Feature pooling over quadrants (Φ_3)
Feature pooling over quadrants (Φ_5)

Understanding the impact of the number of neurons in SCLs and DPLs: The results reported in Fig. 3a clearly shows that the performance of the DHN combining the features from the 1^{st} and 2^{nd} SCL, noted ($\Phi_1 + \Phi_3$), consistently outperforms Φ_1 and Φ_3 alone, which is also confirmed in Fig. 3b. Fig. 3a shows a monotonic increase of classification accuracy with the number of neurons in the 2^{nd} SCL, again confirmed in Fig. 3b. However, the influence of the number of neurons in the 1^{st} DPL on the classification accuracy is more subtle. It appears in Fig. 3b that there exists an optimal number of neurons for that layer, which is around 15 neurons. An explanation of this phenomenon is the fact that DPL learns a low-dimensional linear subspace in which part of the information might be lost if the dimension is too small. Reciprocally, if the embedding space is too big, the following SCL is unable to learn an appropriate representation as in the naive multi-layer sparse coding model proposed in [1].

Fig. 3: Classification accuracy of different 3-layer DHN on the MIT-67 dataset.



(a) Classification accuracy as a function of the number of neurons in the 2^{nd} SCL for different sizes of the 1^{st} DPL (10, 20, and 30 neurons) and using either only Φ_1 or Φ_3 or both.

(b) Classification accuracy as a function of the number of neurons in the 1^{st} DPL for different sizes of the 2^{nd} SCL (200 and 800 neurons) and using either only Φ_1 or Φ_3 or both.

Impact of the number of layers: Table 1 shows that the classification accuracy of a DHN increases when using features from more layers on the MIT-67 dataset. The features extracted from Φ_5 appear beneficial in the MIT-67 but not so in the CIFAR-10. The highest classification accuracy on the MIT-67, 41.4%, is reached when the features of the three SCLs are combined ($\Phi_1 + \Phi_3 + \Phi_5$). For the CIFAR-10 however, the highest accuracy, 79.1%, is achieved using only the features from the 1^{st} and 2^{nd} SCL ($\Phi_1 + \Phi_3$).

This discrepancy is likely due to the difference in sizes of the images used, 32x32 pixels and 100x100 pixels. Such results support the utility of using convolutional architecture and depth pooling, which enables the DHN to successfully exploit both local and global discriminative information, which are necessary for addressing scene recognition problems.

Comparison to other models: On the MIT-67, the DHN shows higher accuracy than the PCANet [5], Deformable Parts models (DPM), Spatial Pyramid Matching (SPM)

Table 1: Classification accuracy when using features from different layers of the DHN on the MIT-67 and CIFAR-10. The features extracted by the DHN are used to train a linear SVM. $\Phi_1 + \Phi_3 + \Phi_5$ denotes the concatenation of features from the three SCLs.

	Features used in the Linear SVM for the Classification				
	Φ_1 (300 neurons)	Φ_3 (1200 neurons)	Φ_5 (3200 neurons)	$\Phi_1 + \Phi_3$	$\Phi_1 + \Phi_3 + \Phi_5$
MIT-67	28.5 %	32.4 %	35.1 %	37.8 %	**41.4**%
CIFAR-10	72.2%	76.6%	61.6%	**79.1%**	74.5%

[13] and Reconfigurable Models (RBoW) [14], as reported in Table 2. It reaches similar accuracy to Hierarchical Matching Pursuit trained on RGB images (HMP-RGB) [3]. However, the combined model (DPM+Gist+SPM) [13] and the Multipath-HMP (M-HMP) [4] still outperform the DHN. Improvements of the architecture of the DHN inspired by the M-HMP may enable it to capture richer features at different scales.

Although on the CIFAR-10 the performance of the DHN are comparable to the single-layer Hebbian (SLH) introduced in [1], it does so with half of the neurons used in the single layer in [1], which increases further its computational and memory efficiency.

Table 2: Evaluation of the DHN against other unsupervised models on (a) MIT-67 and (b) CIFAR-10.

(a) Classification accuracy on the MIT-67

Algorithm	Accuracy
DHN ($\Phi_1 + \Phi_3 + \Phi_5$)	**41.4 %**
SPM [13]	34.4 %
PCANet [5]	34.7 %
RBoW [14]	37.9 %
HMP - RGB [3]	41.8 %
DPM+Gist+SPM [13]	43.1 %
M-HMP [4]	51.2 %

(b) Classification accuracy on the CIFAR-10

Algorithm	Accuracy
DHN ($\Phi_1 + \Phi_3$)	**79.1 %**
Sparse RBM	72.4 %
PCANet [5]	78.7 %
Single-layer Hebbian [1]	79.6 %
Multi-layer K-means [6]	82.0 %
TIOMP-1/T [19]	82.2 %
Multi-Layer NOMP [11]	82.9 %

5 Conclusion

This work introduces the first multi-layer Hebbian network, called DHN, which combines sparse coding and dimensionality reduction. It is the first time a Hebbian network has shown competitive performance at unsupervised features learning for image classification tasks. When evaluated on indoor scene recognition, the DHN achieves higher accuracy than many algorithms, e.g. RBoW. Although the model does not reach the highest accuracy on those benchmarks, it has the major advantage of being trainable online, making it an excellent candidate for learning from unbounded streams of data.

The power and memory efficiency of the architecture proposed might also prove particularly useful for mobile and embedded computing. Recent work [17] already explores potential hardware devices implementing similar principles to those used in the DHN. Although the DHN proves competitive when compared to unsupervised models, it does not compare to models using back-propagation. Future work will explore the introduction of supervision in the form of local learning rules [2] in the DHN.

References

1. Bahroun, Y., Soltoggio, A.: Online representation learning with single and multi-layer Hebbian networks for image classification tasks. In: International Conference on Artificial Neural Networks (To appear) (2017)
2. Baldi, P., Sadowski, P.: A theory of local learning, the learning channel, and the optimality of backpropagation. Neural Networks 83, 51–74 (2016)
3. Bo, L., Ren, X., Fox, D.: Hierarchical matching pursuit for image classification: Architecture and fast algorithms. In: NIPS. vol. 1, p. 6 (2011)
4. Bo, L., Ren, X., Fox, D.: Multipath sparse coding using hierarchical matching pursuit. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 660–667 (2013)
5. Chan, T.H., Jia, K., Gao, S., Lu, J., Zeng, Z., Ma, Y.: PCANet: A simple deep learning baseline for image classification? IEEE Transactions on Image Processing 24(12), 5017–5032 (2015)
6. Coates, A., Ng, A.Y.: Selecting receptive fields in deep networks. In: Advances in Neural Information Processing Systems. pp. 2528–2536 (2011)
7. Cortes, C., Vapnik, V.: Support-vector networks. Machine learning 20(3), 273–297 (1995)
8. Cox, T.F., Cox, M.A.: Multidimensional scaling. CRC press (2000)
9. Hosoya, H., Hyvärinen, A.: Learning visual spatial pooling by strong PCA dimension reduction. Neural computation (2016)
10. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images (2009)
11. Lin, T.h., Kung, H.: Stable and efficient representation learning with nonnegativity constraints. In: Proceedings of the 31st International Conference on Machine Learning (ICML-14). pp. 1323–1331 (2014)
12. Olshausen, B.A., Field, D.J.: Sparse coding with an overcomplete basis set: A strategy employed by V1? Vision research 37(23), 3311–3325 (1997)
13. Pandey, M., Lazebnik, S.: Scene recognition and weakly supervised object localization with deformable part-based models. In: Computer Vision (ICCV), 2011 IEEE International Conference on. pp. 1307–1314. IEEE (2011)
14. Parizi, S.N., Oberlin, J.G., Felzenszwalb, P.F.: Reconfigurable models for scene recognition. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. pp. 2775–2782. IEEE (2012)
15. Pehlevan, C., Chklovskii, D.: A normative theory of adaptive dimensionality reduction in neural networks. In: Advances in Neural Information Processing Systems. pp. 2269–2277 (2015)
16. Pehlevan, C., Chklovskii, D.B.: A Hebbian/anti-Hebbian network derived from online nonnegative matrix factorization can cluster and discover sparse features. In: 2014 48th Asilomar Conference on Signals, Systems and Computers. pp. 769–775. IEEE (2014)
17. Poikonen, J.H., Laiho, M.: Online linear subspace learning in an analog array computing architecture. 16th International Workshop on Cellular Nanoscale Networks and their Applications (CNNA) (2016)
18. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 413–420. IEEE (2009)
19. Sohn, K., Lee, H.: Learning invariant representations with local transformations. In: Proceedings of the 29th International Conference on Machine Learning. pp. 1311–1318 (2012)
20. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–9 (2015)
21. Zhang, S., Wang, J., Tao, X., Gong, Y., Zheng, N.: Constructing deep sparse coding network for image classification. Pattern Recognition 64, 130–140 (2017)