Rotation-Invariant Restricted Boltzmann Machine Using Shared Gradient Filters

Mario Valerio Giuffrida[1,2(✉)] and Sotirios A. Tsaftaris[1,2]

[1] IMT Scuola Alti Studi Lucca, PRIAn, Lucca, Italy
valerio.giuffrida@imtlucca.it
[2] School of Engineering, University of Edinburgh, Edinburgh, UK
s.tsaftaris@ac.ed.uk

Abstract. Finding suitable features has been an essential problem in computer vision. We focus on Restricted Boltzmann Machines (RBMs), which, despite their versatility, cannot accommodate transformations that may occur in the scene. As a result, several approaches have been proposed that consider a set of transformations, which are used to either augment the training set or transform the actual learned filters. In this paper, we propose the *Explicit Rotation-Invariant Restricted Boltzmann Machine*, which exploits prior information coming from the dominant orientation of images. Our model extends the standard RBM, by adding a suitable number of weight matrices, associated with each dominant gradient. We show that our approach is able to learn rotation-invariant features, comparing it with the classic formulation of RBM on the MNIST benchmark dataset. Overall, requiring less hidden units, our method learns compact features, which are robust to rotations.

Keywords: Rotation invariance · Restricted Boltzmann Machine · Explicit invariance · Shared filters

1 Introduction

It is widely known that a crucial problem in image understanding is to find suitable features for the task at hand. Hand-crafted descriptors were able to provide adequate representations, but they rely on specific structures in the scene and could not accommodate certain nuisance factors properly. Hence, extensive efforts in learning image representations have been done in the past years, demonstrating that machine learning approaches are able to outperform hand-crafted descriptors [23]. Examples of learned features are e.g. vocabulary learning [5], sparse coding [15], Gaussian mixture models [1], neural networks [2].

Neural networks (NNs) are graphical models, where nodes in a graph are connected with weighted connections and parameters are determined via optimisation algorithms. The *Restricted Boltzmann Machine* (RBM) has recently gained popularity, mainly because of its applications to deep learning [2,12]. RBM is a generative NN constituted by a bipartite graph, which sides are referred to *visible*

A.E.P. Villa et al. (Eds.): ICANN 2016, Part II, LNCS 9887, pp. 480–488, 2016.
DOI: 10.1007/978-3-319-44781-0_57

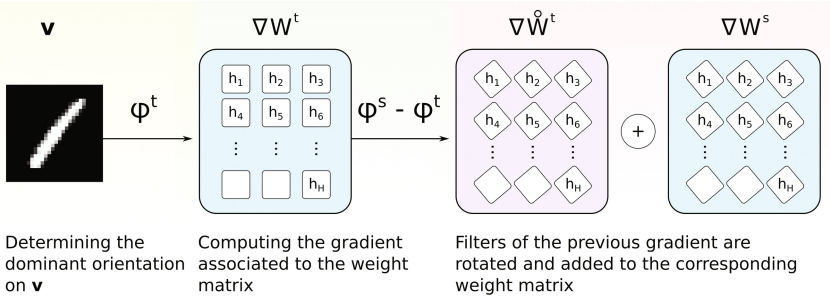

Fig. 1. The dominant orientation φ^t is determined for the provided image and is used to compute the gradient $\nabla W^{(t)}$. The contribution of this gradient is shared amongst the other weight matrices $\nabla W^{(s)}$, $s = 1, 2, \ldots, S$, $t \neq s$, rotating the learned filters by the angle $\varphi^s - \varphi^t$ to generate the $\nabla \mathring{W}^{(t)}$ term.

layer and *hidden layer* respectively. The set of parameters within the RBM are optimised via the *Contrastive Divergence* (CD) algorithm [11]. Although RBMs can achieve satisfactory results [4], their use in shallow networks (namely few layers) cannot accommodate complex variability occurring in the scene [20]. To this end, the *Deep Belief Network* (DBN) was proposed in [14], which is constituted by several stacked RBMs. Albeit DBN have been shown to achieve some translation invariance, they may not well accommodate other nuisance factors (e.g. rotation).

In fact, several modifications of the original RBM formulation have been recently proposed, achieving certain transformation invariance. In [21], a transformation invariant RBM is proposed, where images are subjected to a predefined set of transformations. In [13] an RBM that learns equivariant features is proposed, whereby adding a new variable to be inferred within the hidden units, this variable is then used to rotate learned weights accordingly. In [19], a rotation (invariant) Convolutional RBM is proposed. The marginal probability of RBM is extended with a Markov Random Field, including transformed versions of input images. In [20], an additional step of the backpropagation algorithm used to train DBN is introduced, where the weights are transformed and the entire network is trained again. In [3], the authors propose an RBM where input images are divided into non-overlapping blocks. Then, patches are extracted on SIFT keypoints [18] and subsequently rotated and scaled accordingly. Despite their progress, the aforementioned methods share the following drawbacks: either they are limited to the set of transformations considered within the model, or they involve deep networks in the hope of learning better transformation invariant features [13,20,21], albeit increasing computational demand.

In this paper instead we present the *Explicit Rotation-Invariant Restricted Boltzmann Machine* (ERI-RBM), which can model the nuisance caused by rotated versions of the same pattern, without actually applying any transformation to the data. Our method considers a set of weight matrices

(similar concept as in C-RBM [16]) and each sample is provided to the visible layer with its dominant orientation [3]. This information is used to select a particular weight matrix during the Gibbs sampling to compute gradients of parameters. The contribution given by the new update gradients is shared among the other weight matrices, rotating the filters accordingly [20] (cf. Fig. 1). Experiments on MNIST-rot show superior performance to several baseline benchmarks and a recent method from the literature.

Our contributions are multi-fold: (i) rotation is treated explicitly, without rotating the image patterns, in contrast to for example [21]; (ii) we adopt a shallow model using a limited amount of additional weight matrices, instead of deep architectures [17]; (iii) we share the contribution coming from a weight matrix with the other ones, rotating the learned filters by suitable angles.

This paper is organised as follows. Section 2 describes the proposed Explicit Rotation-Invariant Restricted Boltzmann Machine. In Sect. 3, we present experimental results, whereas Sect. 4 concludes the manuscript.

2 Explicit Rotation-Invariant RBM (ERI-RBM)

In this section, we discuss how to embed the concept of rotation-invariance explicitly in the RBM formulation. Since input patterns are images, we will assume that neurones in the visible layer are arranged in matrix form of size $w \times h = d$, width and height respectively. Each row in the weight matrix W, connecting visible units to hidden units, is a d-dimensional vector. Therefore, each row in W can also be arranged in matrix form of size $w \times h$. Henceforth, we will refer to rows in the weight matrix W as *learned filters* and rows in ∇W as *update filters*, which is the gradient computed during the Contrastive Divergence algorithm.

2.1 Proposed Model

Let Φ be a set of evenly distanced angles $\Phi = \{\varphi_1, \varphi_2, \ldots, \varphi_S\}$, such that for any $i \leq j \implies \varphi_i \leq \varphi_j$. In our model, we augment the number of weight matrices $W \in \mathbb{R}^{H \times V \times S}$, such that every angle φ_s is associated to a matrix $W^{(s)}$. Here, H is the number of hidden units, V the number of visible units, and S is the number of angles. In addition, each weight matrix has an associated bias vector $\boldsymbol{b}^{(s)}$. Hence, we rewrite the energy function characterising the standard Restricted Boltzmann Machine formulation as follows:

$$E(\boldsymbol{v}, \boldsymbol{h}; s) = -\boldsymbol{h}^T W^{(s)} \boldsymbol{v} - \boldsymbol{c}^T \boldsymbol{v} - \left[\boldsymbol{b}^{(s)}\right]^T \boldsymbol{h}, \tag{1}$$

where $W^{(s)}$ is the s-th weight matrix, $\boldsymbol{b}^{(s)}$ is the bias vector for the hidden layer associated to $W^{(s)}$, with $s = 1, 2, \ldots, S$, and $\boldsymbol{c}$ is the bias vector for the visible layer. The index s is uniquely determined on each input image $\boldsymbol{v}$, and will be discussed thoroughly in Sect. 2.2. Because of the modification in (1), all

the equations involved in the CD algorithm have to be rewritten. Specifically, the conditional probabilities become:

$$p(h_k = 1|\boldsymbol{v}; s) = \sigma\left(b_k^{(s)} + \boldsymbol{W}_{\boldsymbol{k},\bullet}^{(\boldsymbol{s})}\boldsymbol{v}\right), \tag{2}$$

$$p(v_j = 1|\boldsymbol{h}; s) = \sigma\left(c_j + \boldsymbol{h}^T\boldsymbol{W}_{\bullet,\boldsymbol{j}}^{(\boldsymbol{s})}\right). \tag{3}$$

During the optimisation algorithm, an image $\boldsymbol{v}$ with dominant orientation φ_s is provided to the Gibbs sampling. After a sufficient number of alternating computations of (2) and (3), the gradient $\nabla W^{(s)}$ can be computed, whose contribution is shared with the remaining matrices in W. To update $\nabla W^{(t)}$, $1 \leq t \leq S$, $t \neq s$, we transform the update filters in $\nabla W^{(s)}$ which are then added to the t-th gradient. Specifically, since we can represent rows in $\nabla W^{(s)}$ as images, they can be rotated by an angle $\theta = \phi_t - \phi_s$. Therefore, we define a new *shared update filter* term $\nabla \mathring{W}^{(t)}$, such that

$$\nabla \mathring{W}^{(t)} = R_\theta(\nabla W^{(s)}) \equiv \begin{pmatrix} R_\theta\left(\boldsymbol{\nabla W}_{\boldsymbol{1},\bullet}^{(\boldsymbol{s})}\right) \\ R_\theta\left(\boldsymbol{\nabla W}_{\boldsymbol{2},\bullet}^{(\boldsymbol{s})}\right) \\ \vdots \\ R_\theta\left(\boldsymbol{\nabla W}_{\boldsymbol{H},\bullet}^{(\boldsymbol{s})}\right) \end{pmatrix}. \tag{4}$$

where $R_\theta = [\cos\theta \ -\sin\theta; \sin\theta \ \cos\theta]$ defines the 2D rotation matrix by an angle θ. This operation may generate filters bigger than the input layers and we crop them such that the filter size remains $w \times h$. At this point, the final expression for the gradient $\nabla W^{(s)}$ is updated as follows:

$$\nabla W^{(s)} := \nabla W^{(s)} + \nabla \mathring{W}^{(s)}. \tag{5}$$

Note that (5) will be utilised within the Stochastic Gradient Descent step of the CD algorithm. Therefore, $\nabla W^{(s)}$ will be multiplied by a learning rate η that typically has values set in the order of 10^{-3} (further details are discussed in [10]). Hence, any side effects originating from pixel interpolation are minimised, precisely because of the small η. Gradients $\boldsymbol{\nabla b}^{(s)}$ are computed as described in [11], using samples $\boldsymbol{v}$ with the associated dominant orientation φ_s.

2.2 Finding the Dominant Angle and Corresponding s Index

Each image $\boldsymbol{v}$ is associated to an angle φ_s, determined by the histogram of oriented gradients from $\boldsymbol{v}$ [6]. Derivatives along the x and y directions are computed and the angle of each gradient vector can be determined. All the vectors are accumulated into a histogram with S bins and the angle ψ with the highest frequency is found. Formally, the index $s = \text{argmax}_j \varphi_j$, such that $\varphi_j \leq \psi$, $\varphi_j \in \boldsymbol{\Phi}$. Figure 2 shows graphically those steps: from the original image pattern (a), derivatives are computed using Sobel filters (b). Subsequently, we build the weighted histogram of oriented gradients and the angle with the highest

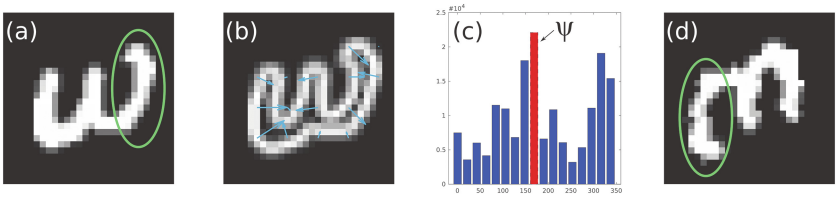

Fig. 2. Computation of the dominant orientation for a sample image taken from the MNIST dataset: (a) original sample, (b) gradients of the image, (c) histogram of oriented gradients with highlighted mode ψ, (d) sample rotated by ψ degree. The region marked by a green ellipse corresponds to the same portion of the number 3 in the original and rotated image. Observe the differences due to image interpolation introduced during rotation.

frequency ψ is selected (c). We highlight in red the 9-th bin of the histogram, hence $s = 9$ for the illustrated example. In (d) we report a rotated version of the sample image by ψ degree to show the deleterious effect of image interpolation.

Since strong edges near image boundaries may bias the estimation of the dominant gradient, the magnitude of the corresponding vectors is weighted with a Gaussian kernel, with $\sigma = \frac{min\{w,h\}}{5}$ (width and height of $\boldsymbol{v}$ respectively), such that central gradients contribute more than those at the boundaries. (We found this value covers evenly the entire image without exceeding its size.)

3 Experimental Results

Setup: We used the MNIST-rot dataset[1] [14], containing 10,000 images for training, 2,000 for validation, and 50,000 for testing. This dataset is derived from the MNIST dataset, where samples were rotated by random angles. To enable comparison with other methods, for consistency, we kept this dataset splitting, and we did not perform cross-validation (that could have provided variances for statistical analysis). Since each image contains several non-zero entries close to 0, we threshold them at a value $\tau = 0.3$. We compare ERI-RBM with several informative baselines and a recent invariant method. *Classical RBM:* We trained a standard Bernoulli Restricted Boltzmann Machine and compared results with our Explicit Rotation-Invariant RBM. *Dominant RBM (D-RBM):* We built a simplified model that learns an RBM for each dominant orientation, splitting the training set into S partitions, associated to a different RBM (i.e., we have S independent RBMs). *Oriented RBM (O-RBM):* We pre-process the dataset by aligning all images according to their dominant orientation to a reference orientation and train a single RBM. *TI-RBM*: We also compared with the method in [21], using the authors implementation[2]. Extracted features are provided to the following classifiers: linear and RBF SVM [22], softmax [9], and K-NN [7].

[1] Available at http://www.iro.umontreal.ca/~lisa/twiki/bin/view.cgi/Public/DeepVsShallowComparisonICML2007.

[2] Available at https://github.com/kihyuks/icml2012_tirbm.

Table 1. Testing accuracies of standard RBM, Dominant RBM, Oriented RBM, TI-RBM [21], and our proposed ERI-RBM.

	RBF SVM $\mathbf{C{=}10, \gamma = 0.1}$	**Linear SVM** $\mathbf{C{=}0.1}$	**Softmax**	**K-NN** $\mathbf{K = 3}$
RBM (H = 100)	87.37 %	59.27 %	57.80 %	82.69 %
D-RBM (H = 100, S = 4)	83.44 %	58.95 %	56.80 %	78.84 %
D-RBM (H = 100, S = 9)	79.18 %	53.62 %	50.76 %	73.56 %
D-RBM (H = 100, S = 18)	69.84 %	49.20 %	46.58 %	63.61 %
O-RBM (H = 100 S = 18)	87.37 %	58.99 %	57.80 %	82.69 %
ERI-RBM (H = 100, S = 4)	78.49 %	60.27 %	58.31 %	74.97 %
ERI-RBM (H = 100, S = 9)	91.27 %	74.87 %	73.02 %	88.48 %
ERI-RBM (H = 100, S = 18)	**92.08** %	**77.69** %	**75.84** %	**89.34** %
TI-RBM [21] **(H = 100, S = 18)**	80.63 %	69.10 %	68.20 %	73.60 %

Parameters: We set the number of hidden units to $H = 100$, while progressively increased the number of bins S, used to generate the histogram of orientations. Following the instructions in [10], we set the learning rate $\eta = 10^{-3}$, the Contrastive Divergence algorithm is iterated up to 200 epochs, and a constant momentum $\alpha = 0.9$ was used. The parameters for SVM were found using logarithmic grid search and best values are reported in Table 1. We set arbitrary $K = 3$ for the K-NN, using the Euclidean distance as metric. For TI-RBM [21], a set of $K = S$ transformations are considered, which is each associated with an array of H hidden units, while a single weight matrix W is considered. The final representation used during inference is obtained by max-pooling. To make the comparison to ERI-RBM fair, for TI-RBM the sparsity term was disabled, and we set the number of hidden units to $H = 100$.

Discussion: We report our results in Table 1 and we noticed that nonlinear SVM gave the best performance in all the cases. The baseline is given by RBM with an accuracy of 87 %. Tests using D-RBM show a gradual loss of accuracy as the number of dominant orientations S is increased. This behaviour can be attributed to the lack of information sharing amongst the RBMs, since they were each trained independently with less data (per RBM). Overall, our proposed model outperforms the baseline RBM ($S \geq 9$). At $S = 4$, ERI-RBM has a loss of performance, because of the coarse quantization of the 2π space: angles 0°, 90°, 180°, and 270° will have orthogonal rotations when shared update filters are computed for neighbour matrices, causing the propagation of sharp rotations that do not contribute much. As the number of S increases, ERI-RBM has a +13 % of improvement, showing that our model is able to learn rotation-invariant features. This is also displayed in Fig. 3, showing learned filters when $S = 9$. O-RBM shows no improvement compared to RBM, demonstrating that the contribution provided by the shared update filters increases the

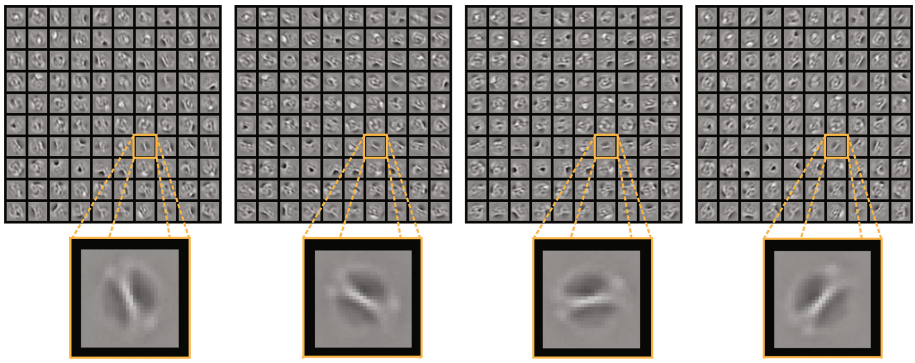

Fig. 3. Filters learned by our ERI-RBM at $S = 9$. We highlight a filter that appears at rotations 0°, 40°, 80°, and 120°, showing that our model learns rotation-invariant filters. The remaining weight matrices are omitted for brevity.

discriminative power of the final representation. Note that we also trained classical RBM with $H = 1000$, noticing an improvement of 2 %, still lower than ERI-RBM. Finally, using the same experimental setup, ERI-RBM outperformed [21] by +12 % in testing accuracy. (These results are different from those reported in [21] since sparsity is not present and we used less units.) Our approach does rely on the determination of orientation, which could be seen as a limitation. Preliminary results (not shown for brevity), obtained by artificially perturbing the orientation estimate, show that we are tolerant to such errors up to ±4 bins off on the original estimate. This remains to be confirmed in images with cluttered background.

4 Conclusions

In this paper we proposed the *Explicit Rotation-Invariant Restricted Boltzmann Machine* (ERI-RBM). Current approaches do not address the problem of rotation-invariance directly, but use a predefined set of transformations to transform either the input images [19,21] or the learned filters [13,20]. We were inspired by these approaches to modify the RBM learning process, such that to learn invariant features without taking into account all possible transformations, which is demanding and may propagate noise due to pixel interpolations.

Our ERI-RBM utilises the dominant gradient of input images in order to select the best set of filters to optimise. We find the corresponding gradients efficiently and update the filters in a process where information is shared across the different filters, minimising thus any effects of interpolation. Overall, our model learns rotation-invariant features and achieves an accuracy of 92 % in the MNIST-rot dataset. Comparisons with several baselines and approaches from the literature showed superior performance in a common experimental setup. Moreover, comparing to the deep architecture of [8] and the results on MNIST-rot,

ERI-RBM reached similar performance using just 100 of hidden units compared to the 500 in [8]. In conclusion, ERI-RBM is able to learn rotation-invariant features in an unsupervised fashion, with a reduced number of hidden units, within a shallow network.

Acknowledgements. We thank NVIDIA corporation for providing us a Titan X GPU.

References

1. Agarwal, A., Triggs, B.: Hyperfeatures – multilevel local coding for visual recognition. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 30–43. Springer, Heidelberg (2006)
2. Arel, I., Rose, D.C., Karnowski, T.P.: Deep machine learning - a new frontier in artificial intelligence research. IEEE Comput. Intell. Mag. **5**(4), 13–18 (2010)
3. Cheng, D., Sun, T., Jiang, X., Wang, S.: Unsupervised feature learning using Markov deep belief network. In: 2013 IEEE International Conference on Image Processing, pp. 260–264, No. 20120073110053. IEEE (2013)
4. Coates, A., Arbor, A., Ng, A.Y.: An analysis of single-layer networks in unsupervised feature learning. In: AISTATS, pp. 215–223 (2011)
5. Csurka, G., Dance, C.R., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: Proceedings of the ECCV International Workshop on Statistical Learning in Computer Vision, pp. 59–74 (2004)
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of the IEEE CVPR, vol. 1, pp. 886–893 (2005)
7. Dasarathy, B.: Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, Los Alamitos (1991)
8. Gens, R., Domingos, P.M.: Deep symmetry networks. In: NIPS, pp. 2537–2545. Curran Associates, Inc. (2014)
9. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer Series in Statistics, vol. 1, 2nd edn. Springer, New York (2009)
10. Hinton, G.: A Practical Guide to Training Restricted Boltzmann Machines, 2nd edn. Springer, Berlin (2012)
11. Hinton, G.E.: Training products of experts by minimizing contrastive divergence. Neural Comput. **14**(8), 1771–1800 (2002)
12. Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. Neural Comput. **18**(7), 1527–1554 (2006)
13. Kivinen, J.J., Williams, C.K.I.: Transformation equivariant boltzmann machines. In: Honkela, T. (ed.) ICANN 2011, Part I. LNCS, vol. 6791, pp. 1–9. Springer, Heidelberg (2011)
14. Larochelle, H., Erhan, D., Courville, A., Bergstra, J., Bengio, Y.: An empirical evaluation of deep architectures on problems with many factors of variation. In: Proceedings of the 24th ICML, pp. 473–480 (2007)
15. Lee, H., Battle, A., Raina, R., Ng, A.Y.: Efficient sparse coding algorithms. In: Advances in Neural Information Processing Systems, pp. 801–808 (2006)
16. Lee, H., Ekanadham, C., Ng, A.Y.: Sparse deep belief net model for visual area V2. In: Advances in Neural Information Processing Systems, pp. 873–880 (2008)
17. Lee, H., Grosse, R., Ranganath, R., Ng, A.Y.: Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: ICML (2009)

18. Lowe, D.G.: Object recognition from local scale-invariant features. In: ICCV (1999)
19. Schmidt, U., Roth, S.: Learning rotation-aware features: from invariant priors to equivariant descriptors. In: Proceedings of the IEEE CVPR, pp. 2050–2057 (2012)
20. Shou, Z., Zhang, Y., Cai, H.J.: A study of transformation-invariances of deep belief networks. In: IJCNN, pp. 1–8. IEEE (2013)
21. Sohn, K., Lee, H.: Learning invariant representations with local transformations. In: Proceedings of the 29th ICML, pp. 1311–1318 (2012)
22. Vapnik, V.: Statistical Learning Theory. Wiley, New York (1998)
23. Wei, X., Phung, S.L., Bouzerdoum, A.: Visual descriptors for scene categorization: experimental evaluation. Artif. Intell. Rev. **45**(3), 1–36 (2015)