# Gated Boltzmann Machine for Recognition under Occlusion

**Yichuan Tang**
Dept. of Computer Science, Univ. of Toronto
6 King's College Rd.
Toronto, Ontario, CANADA
tang@cs.toronto.edu

Unconstrained real world environments are often full of clutter. Therefore, robustness to occlusion is vital for any artificial recognition system. Recently, the Deep Boltzmann Machine (DBM) has been shown to be good at generative modeling and recognition of visual objects. In this work we develop an extension to the DBM framework to make the system more accurate when recognizing handwritten digits under partial occlusion. The key is the introduction of additional indicator random variables which specify where in the image to ignore the occluder. The new model is still a Boltzmann machine as some extra terms are added to the DBM energy function. During inference, the model tries to figure out what are the occluder and the "object" given an occluded image. In addition, we can easily transfer the learned occluder model to other DBMs learned on different types of data, e.g. faces or objects.

## 1 Deep Boltzmann Machines

A Deep Boltzmann Machine (DBM) consists of several layers of binary stochastic nodes including D visible nodes $\mathbf{v} \in \{0,1\}^D$ and hidden nodes $\mathbf{h}^j \in \{0,1\}^{N_j}$, where j is the layer label. With only connections between adjacent layers, a two layer DBM can be defined by an energy function:

$$E_{DBM}(\mathbf{v}, \mathbf{h}^1, \mathbf{h}^2) = -\mathbf{v}^\mathsf{T}\mathbf{W}^1\mathbf{h}^1 - (\mathbf{h}^1)^\mathsf{T}\mathbf{W}^2\mathbf{h}^2 \tag{1}$$

The probability of any given state $\{\mathbf{v}, \mathbf{h}^1, \mathbf{h}^2\}$ can be obtained by exponentiating and dividing by the normalization constant. [1, 2] describe ways to train the DBM to model high dimensional visual data as well as achieve good recognition results.

During inference, $\mathbf{v}$ is clamped and the posterior $p(\mathbf{h}|\mathbf{v})$ is approximated using mean-field updates. However, this is not desirable when there are significant noise in $\mathbf{v}$ as the entire input, including the noise, is conditioned upon in $p(\mathbf{h}|\mathbf{v})$.

## 2 Denoising Gated Boltzmann Machines

To provide robustness for situations with occlusion, we introduce the Denoising Gated Boltzmann Machine (DGBM). It is a modified version of the DBM where we have added two additional sets of variables to help "explain" the occluder in an image. Figure 1 shows the architecture of the DGBM.

The new energy function is defined as[1]:

$$E_{DGBM} = E_{DBM}(\mathbf{v}, \mathbf{h}^1, \mathbf{h}^2) - \boldsymbol{\psi}^\mathsf{T}\mathbf{U}\mathbf{g} + \sum_i^D \gamma_i \psi_i \log(1 + (v_i - \tilde{v}_i)^2) \tag{2}$$

In the DGBM, the difference is in the introduction of $\boldsymbol{\psi} \in \{0,1\}^D$ and $\tilde{\mathbf{v}} \in \{0,1\}^D$ and their 3rd order interactions with $\mathbf{v}$. During inference, $\tilde{\mathbf{v}}$ is observed and $\mathbf{v}$ becomes latent. When $\psi_i = 1$,

[1]The biases are omitted for clarity of presentation.

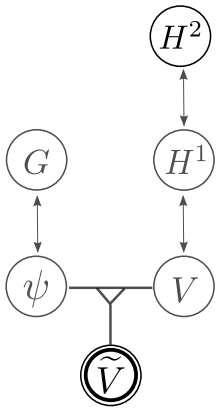


Figure 1: The network diagram of a DGBM.

the model[2] strongly prefers that $v_i \approx \tilde{v}_i$. When $\psi_i = 0$, the *log1p(·)* penalty term is gated off, and the DBM is free to fantasize about what is behind the occluder. Therefore, the role of $\boldsymbol{\psi}$ serves as an indicator for where to ignore any possible occluders. In addition, the DGBM also contains an additional RBM with weights $\mathbf{U}$ and hidden nodes $\mathbf{g}$ to model the structure of $\boldsymbol{\psi}$.

### 2.1 Inference

The posterior of interest is $p(\mathbf{v}, \boldsymbol{\psi}|\tilde{\mathbf{v}})$, or the distribution of the unoccluded image and the occluder given $\tilde{\mathbf{v}}$. We can sample from this posterior via alternate Gibbs sampling of $p(\mathbf{v}, \boldsymbol{\psi}|\mathbf{h}^1, \mathbf{g}, \tilde{\mathbf{v}})$, $p(\mathbf{g}|\boldsymbol{\psi})$, and $p(\mathbf{h}^1|\mathbf{v})$. $p(\mathbf{h}^1|\mathbf{v})$ is sampled by performing an up-pass through the DBM.

Due to the form of the energy function in eq. 2, there are no interactions between $v_i$, $\tilde{v}_j$, and $\psi_k$, when $i \neq j \neq k$. This means that when we sample $p(\mathbf{v}, \boldsymbol{\psi}|\tilde{\mathbf{v}})$, combinations of $\{v_i, \psi_i\}$ can be sampled independently as a multinomial with 4 different states. The energy of the 4 states are:

$$
\begin{aligned}
&E(v_i = 0, \psi_i = 0) = 0 \\
&E(v_i = 0, \psi_i = 1) = \gamma_i \log(1 + \tilde{v}_i^2) - \sum_k U_{ik} g_k \\
&E(v_i = 1, \psi_i = 0) = -\sum_j W_{ij}^1 h_j^1 \\
&E(v_i = 1, \psi_i = 1) = \gamma_i \log(1 + (1 - \tilde{v}_i)^2) - \sum_k U_{ik} g_k - \sum_j W_{ij}^1 h_j^1
\end{aligned}
$$

We can then sample easily from the probability defined over these 4 states, e.g.

$$
p(v_i = 0, \psi_i = 0) = \frac{\exp^{-E(v_i=0,\psi_i=0)}}{\sum_{v_i,\psi_i} \exp^{-E(v_i,\psi_i)}} \tag{3}
$$

### 2.2 Learning

Given unlabeled images $\{(\mathbf{v}_1), \ldots, (\mathbf{v}_N)\}$, we have a corresponding occluded image as well as the "mask" of the occluder. This results in an enlarged dataset of $\{(\mathbf{v}_1, \tilde{\mathbf{v}}_1, \boldsymbol{\psi}_1), \ldots, (\mathbf{v}_N, \tilde{\mathbf{v}}_N, \boldsymbol{\psi}_N)\}$ Figure 2 shows some examples from this enlarged dataset. We perform learning by maximizing the average log-likelihood of the enlarged data set: $\max_\theta \frac{1}{N} \log p(\mathbf{v}, \tilde{\mathbf{v}}, \boldsymbol{\psi})$.

Like other undirected graphical models, the gradient of the log-likelihood $l(\theta)$ is given as:

$$
\frac{\partial l(\theta)}{\partial \theta} = -\Big\langle \frac{\partial E_{DGBM}}{\partial \theta} \Big\rangle_{data} + \Big\langle \frac{\partial E_{DGBM}}{\partial \theta} \Big\rangle_{model} \tag{4}
$$

[2] $\gamma_i$ is a parameter that is typically positive and large after learning.

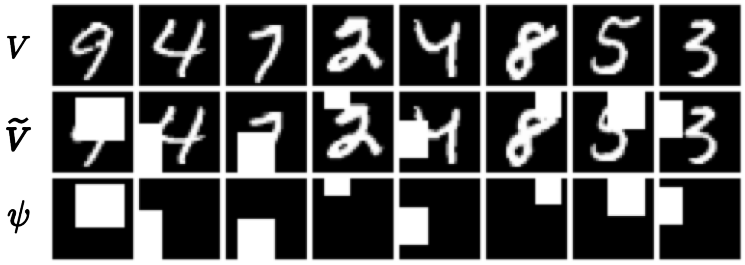


Figure 2: Examples of training data for the DGBM.

Maximum likelihood learning is intractable due to the multiple hidden layers of the DBM. We avoid this problem by using variational learning and approximate the posterior over the hidden layers of the DBM with a factorial distribution. Mean-field fixed-point equations are run to estimate the data-dependent expectations. For the model dependent expectations we use Persistent Contrastive Divergence (PCD) [4].

## 3 Experiments

We performed experiments on the MNIST handwriting dataset with rectangular occluders. To generate the occluded images, we randomly placed blocks (with their width and height varying from 7 to 16 pixels randomly) within the 28 by 28 MNIST digits. We used a DGBM composed of a [784 500 1000] DBM and an additional 300 hidden $\mathbf{g}$ nodes. The DBM was pretrained generatively using variational learning with recognition weights (as described in [2]) for 200 epochs on the regular MNIST data. The RBM defined by weights $\mathbf{U}$ was pretrained using PCD on $\psi$ for 100 epochs. After composing to form the DGBM, we trained for an additional 50 epochs to learn the parameters of the DGBM except for the original DBM weights, which are already a very good model of the digits.

### 3.1 Qualitative Visualizations

For denoising, we used the inference procedure described in section 2.1. In particular, we clamp on $\tilde{\mathbf{v}}$ and run 300 iterations of alternate Gibbs sampling. While Gibbs sampling do not seek to find the mode of the posterior, it nevertheless achieves good results. Figure 3 shows the denoising of test cases.

### 3.2 Recognition Errors

To see how well a DBM would perform at the problem of recognition, we added occluders to the entire MNIST training and testing datasets. A [784 500 1000] DBM was formed by greedy pretraining and generatively fine-tuned for an additional 200 epochs, using recognition weights. To fine-tune further and treating the DBM as a discriminative classifier, we followed the method described in [1]. The activity of $H^2$ layer nodes after 25 iterations of mean-field iterations conditioned on the visible inputs are found for all training data. We then concatenated the $H^2$ activities with the input $V$ and treated these activations as the new input. 3 more hidden layers were added above this new augmented input with the last layer being the softmax output. This entire unrolling of the DBM is used to fully leverage the ability of the DBM to combine top-down and bottom-up predictions.

Following the same strategy for the DGBM, we performed our denoising algorithm using 300 iterations of Gibbs sampling and saved the activity of the $H^2$ layer nodes. We concatenated it with the noisy input data to form a 1784 dimensional input and added 2 additional layers before the output softmax. Figure 4 shows the discriminative DGBM network structure.

For both the DBM and DGBM, backpropagation is used to calculate the gradients and nonlinear Conjugate gradient optimization was used on minibatches of size 1000 for 30 epochs. The test error for the DBM on MNIST with occluder is 7.49% while we achieved a lower error of 6.53% with the

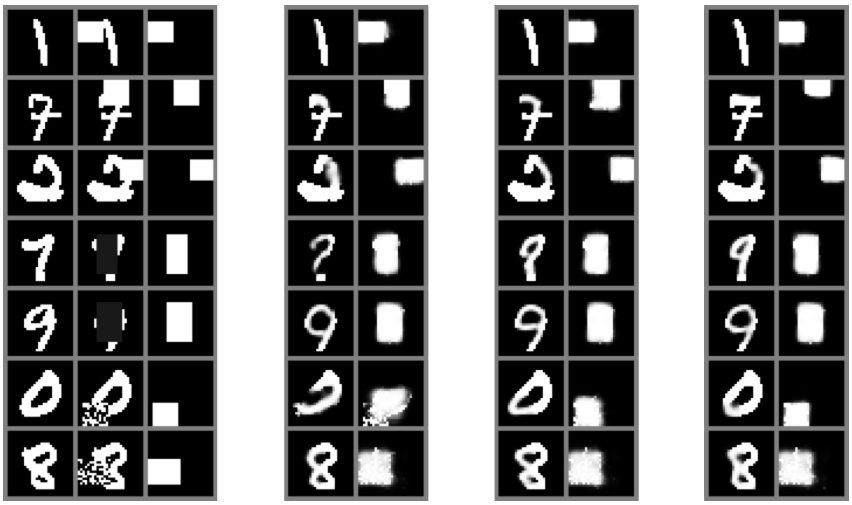

Figure 3: Denoising results. The first 3 columns are the ground truth $\mathbf{v}$, $\tilde{\mathbf{v}}$, and $\boldsymbol{\psi}$, respectively. Subsequent columns show $p(\mathbf{v}|\tilde{\mathbf{v}})$ and $p(\boldsymbol{\psi}|\tilde{\mathbf{v}})$ at iterations of 50, 100 and 300. The process generalizes nicely to block occluders with different appearances.

DGBM. As a comparison, [3] reported an error rate of 8.39% using a discriminative Deep Belief network with sparse connections.

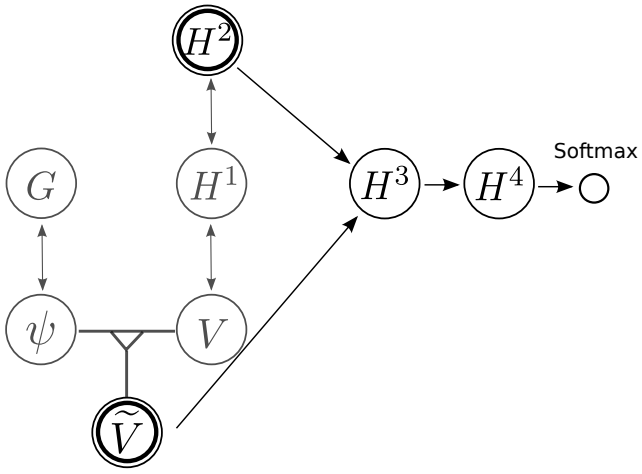


Figure 4: The network diagram of a DGBM for discrimination, $H^3$ and $H^4$ has the same size as $H^1$ and $H^2$, respectively.

## References


[1] R. Salakhutdinov and G. Hinton. Deep Boltzmann machines. In *Proceedings of the Intl. Conf. on Artificial Intelligence and Statistics*, volume 5, pages 448–455, 2009.

[2] Ruslan Salakhutdinov and Hugo Larochelle. Efficient learning of deep boltzmann machines. *Proc. of the 13-th International Conference on AISTATS*, 2010.

[3] Yichuan Tang and Chris Eliasmith. Deep networks for robust visual recognition. In *International Conference on Machine Learning*, 2010.

[4] T. Tieleman. Training restricted boltzmann machines using approximations to the likelihood gradient. In *Intl. Conf. on Machine Learning*, volume 307, pages 1064–1071, 2008.