Robust Boltzmann Machines for Recognition and Denoising

Yichuan Tang
University of Toronto
tang@cs.toronto.edu

Ruslan Salakhutdinov
University of Toronto
rsalakhu@cs.toronto.edu

Geoffrey Hinton
University of Toronto
hinton@cs.toronto.edu

Abstract

While Boltzmann Machines have been successful at unsupervised learning and density modeling of images and speech data, they can be very sensitive to noise in the data. In this paper, we introduce a novel model, the Robust Boltzmann Machine (RoBM), which allows Boltzmann Machines to be robust to corruptions. In the domain of visual recognition, the RoBM is able to accurately deal with occlusions and noise by using multiplicative gating to induce a scale mixture of Gaussians over pixels. Image denoising and inpainting correspond to posterior inference in the RoBM. Our model is trained in an unsupervised fashion with unlabeled noisy data and can learn the spatial structure of the occluders. Compared to standard algorithms, the RoBM is significantly better at recognition and denoising on several face databases.

1. Introduction

Recognition algorithms often break down when solving real world problems. Examples include trying to recognize a face of a person who is drinking from a red coffee mug or trying to find an object partially occluded by a stack of papers. In both cases, the appearance of the occluders should not affect the recognition of the objects of interest, yet many algorithms are significantly influenced by their appearance.

Typical approaches for dealing with occluders are to use an architecture which is engineered to be robust against occlusion and/or to augment the training set with noisy examples. Local descriptors, such as SIFT [11] and Convolutional Neural Nets [9] are examples of such engineered architectures. There are, however, some drawbacks to these approaches. For SIFT and Convolutional Nets, hyperparameters such as the descriptor window size and local filter size need to be specified. Augmenting the training set requires the ability to synthetically generate corruptions, which is challenging for shadows, specular reflections and occlusion by unknown objects.

This paper describes an alternative unsupervised approach that learns to distinguish between corrupted and uncorrupted pixels and to find useful latent representations of both that lead to improved object discrimination. The family of Boltzmann Machine models have been shown to give good results on the facial expression [15] and speech recognition tasks [14]. We present a novel model that allows Boltzmann Machines to be robust to corruptions in the data. Building on a similar model for binary data [21], our model uses multiplicative gating to induce a scale mixture of two Gaussian distributions over the data variables. Furthermore, our framework can successfully learn the *statistical structure* of the noise and occluders without explicit supervision. Our model has several key advantages:

- Multiplicative gating allows for the presence of novel occluders with exotic appearances.
- The structure of the occluders and noise statistics can be learned from the data in an unsupervised fashion.
- Completely automated image inpainting and denoising correspond to posterior inference in the model.

Generative image models with occlusion have been well studied in the vision and machine learning literature [7, 26]. Recently, models involving Restricted Boltzmann Machines have also been applied to image segmentation [17] and foreground-background modeling [3]. Compared to the above work, the fully undirected nature of our model facilitates efficient inference. Face recognition under occlusion has also been explored in [27, 29, 6]. Zhou et al. used an MRF to model contiguous occlusion [29]. However, their model is not as flexible since its parameters are not learned from data.

2. The Model

The Robust Boltzmann Machine (RoBM) is an undirected graphical model with three components. The first is a Gaussian Restricted Boltzmann Machine (GRBM) modeling the density of the noise-free or "clean" data. The second is a Restricted Boltzmann Machine (RBM) modeling the structure of the occluder/noise. The RoBM also contains a multiplicative gating mechanism which allows it to be robust to unexpected corruptions of the observed variables.

978-1-4673-1228-8/12/$31.00 ©2012 IEEE

We briefly review the RBM and GRBM before describing the RoBM in detail.

2.1. Restricted Boltzmann Machines

A Restricted Boltzmann Machine (RBM) is a type of Markov Random Field, or an undirected graphical model that has a bipartite structure with two sets of binary stochastic nodes: the visible $\mathbf{v} \in \{0,1\}^{N_v}$ and hidden $\mathbf{h} \in \{0,1\}^{N_h}$ layer nodes [18]. The RBM has visible to hidden connections but no intra-layer connections. For any configuration of the nodes, we can define an energy function as:

$$E_{RBM}(\mathbf{v},\mathbf{h};\theta) = -\sum_{i}^{N_v} b_i v_i - \sum_{j}^{N_h} c_j h_j - \sum_{i,j}^{N_v,N_h} W_{ij} v_i h_j,$$

where $\theta = \{\mathbf{W},\mathbf{b},\mathbf{c}\}$ are the model parameters. The probability distribution of the configuration $\{\mathbf{v},\mathbf{h}\}$ is:

$$p(\mathbf{v},\mathbf{h};\theta) = \frac{p^*(\mathbf{v},\mathbf{h})}{\mathcal{Z}(\theta)} = \frac{\exp^{-E(\mathbf{v},\mathbf{h})}}{\mathcal{Z}(\theta)}, \tag{1}$$

where we have used $p^*(\cdot)$ to represent the unnormalized probability distribution and $\mathcal{Z}(\theta) = \sum_{\mathbf{v},\mathbf{h}} \exp^{-E(\mathbf{v},\mathbf{h})}$ is the normalization constant. There is a good reason to use RBMs for image modeling. Unlike directed models, an RBM's conditional distribution over hidden nodes is factorial and very easy to compute.

When the data are real valued, the Gaussian RBM (GRBM) [4] can be used for modeling. The GRBM has been successfully applied to tasks including image classification, video action recognition, and speech recognition [10, 8, 22, 14]. The GRBM can be viewed as a mixture of diagonal Gaussians with shared parameters, where the number of mixture components is exponential in the number of hidden nodes and the mixing proportions of the components are defined by marginalizing out the visible nodes from the joint distribution. Its energy is given by:

$$E_{GRBM}(\mathbf{v},\mathbf{h}) = \frac{1}{2}\sum_i \frac{(v_i - b_i)^2}{\sigma_i^2} - \sum_j c_j h_j - \sum_{ij} W_{ij} v_i h_j,$$

The conditional distributions needed for inference and generation are given by:

$$p(h_j = 1|\mathbf{v}) = \frac{1}{1+\exp(-\sum_i W_{ij} v_i - c_j)} \tag{2}$$

$$p(v_i|\mathbf{h}) = \mathcal{N}(v_i|\mu_i, \sigma_i^2) \tag{3}$$

$$\text{where} \quad \mu_i = b_i + \sigma_i^2 \sum_j W_{ij} h_j \tag{4}$$

2.2. Robust Boltzmann Machines

The GRBM is not robust to noise as it assumes a diagonal Gaussian as its conditional distribution over the visible nodes. This means that the log probability assigned to a noisy outlier would be very low and classification accuracy tends to be poor for noisy, out-of-sample test cases. The RoBM solves this problem by using gating at each visible node, inducing a scale mixture of two Gaussians. Its energy is obtained by combining gating terms involving v_i, s_i, and $\tilde{v}_i$, an RBM of binary indicator variables s_i, a GRBM with real-valued variables v_i, and a Gaussian noise model of $\tilde{v}_i$:

$$\begin{aligned} E_{RoBM}(\mathbf{v},\tilde{\mathbf{v}},\mathbf{s},\mathbf{h},\mathbf{g}) &= \frac{1}{2}\sum_i \frac{\gamma_i^2}{\sigma_i^2} s_i (v_i - \tilde{v}_i)^2 \\ &- \sum_i d_i s_i - \sum_k e_k g_k - \sum_{i,k} U_{ik} s_i g_k \\ &+ \frac{1}{2}\sum_i \frac{(v_i - b_i)^2}{\sigma_i^2} - \sum_j c_j h_j - \sum_{ij} W_{ij} v_i h_j \\ &+ \frac{1}{2}\sum_i \frac{(\tilde{v}_i - \tilde{b}_i)^2}{\tilde{\sigma}_i^2}. \end{aligned} \tag{5}$$

In the above energy function, the first line is the gating interaction term involving s_i, v_i, and $\tilde{v}_i$. It allows $\tilde{v}_i$ to be very different from v_i when $s_i = 0$. γ_i^2 regulates the coupling between v_i and $\tilde{v}_i$ when $s_i = 1$. The second line is the energy function of the RBM modeling the structure/correlations of the noise indicators $\mathbf{s}$. The third line is the energy function of the GRBM modeling "clean" data $\mathbf{v}$. The last line in the above energy function specifies the noise distribution: $\tilde{b}_i$ is the mean of the noise and $\tilde{\sigma}_i^2$ is the variance of the noise. In particular, if the model estimates that the i-th node is corrupted with noise ($s_i = 0$), then $\tilde{v}_i \sim \mathcal{N}(\tilde{v}_i|\tilde{b}_i;\tilde{\sigma}_i^2)$.

Fig. 1 shows the graphical model of the RoBM model. Filled triangles emphasize that s_i can dynamically change the weight between v_i and $\tilde{v}_i$. Fig. 2 shows how an RoBM model should decompose an occluded face. Note that only $\tilde{\mathbf{v}}$ is observed and the RoBM model uses its prior over face images to infer the unoccluded face and the occluding shape.

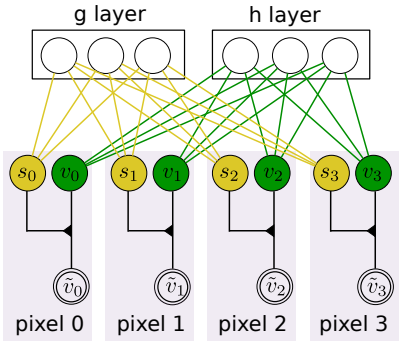

Figure 1. Graphical model of the Robust Boltzmann Machine. Filled triangles indicate gating of the connection between v_i and $\tilde{v}_i$ by s_i. The yellow connections are the weights of the RBM while the green connections are the weights of the GRBM. Each pixel is modeled by three random variables: v_i, $\tilde{v}_i$ and s_i. Best viewed in color.

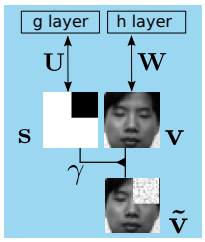

Figure 2. The Robust Boltzmann Machine with real images demonstrating its latent representations. $\tilde{\mathbf{v}}$ is observed, while the model uses the higher layer RBMs to separate out the clean face and the occluder/noise. Best viewed in color.

Properties of the model

The motivation for using the RoBM is to achieve better generalization by eliminating the influence of corrupted pixels. The gating serves as a buffer between what is observed ($\tilde{v}_i$) and what is preferred by the GRBM (v_i). When $\tilde{v}_i$ is corrupted, RoBM can still set v_i to the noiseless value while turning off s_i. If the RBM model of $\mathbf{s}$ assigns equal energies for both states of s_i, then no data penalty costs would be incurred by the corruption to $\tilde{v}_i$.[1]

Robust Statistics, such as the M-estimator [5], use loss functions which do not increase super-linearly. For fitting parametric mixture models, robustness is provided by using a heavy-tailed distribution for the likelihood function of each component [20]. The RoBM model is also a robust mixture model with a scale mixture of two Gaussians over the observed $\tilde{\mathbf{v}}$. To see this, we can formulate RoBM as a mixture model with $2^{N_h+N_g}$ components: $p(\tilde{\mathbf{v}}) = \sum_{\mathbf{h},\mathbf{g}} p(\tilde{\mathbf{v}}|\mathbf{h},\mathbf{g})p(\mathbf{h},\mathbf{g})$, where each component's likelihood function is factorial, $p(\tilde{\mathbf{v}}|\mathbf{h},\mathbf{g}) = \prod_i p(\tilde{v}_i|\mathbf{h},\mathbf{g})$. It can be shown that

$$p(\tilde{v}_i|\mathbf{h},\mathbf{g}) = \prod_i \Big\{ \pi_i \mathcal{N}(\tilde{v}_i|\tilde{b}_i;\tilde{\sigma}_i^2) + (1-\pi_i)\mathcal{N}(\tilde{v}_i|\mu_i^{new};\frac{\sigma_i^2\tilde{\sigma}_i^2}{\tilde{\sigma}_i^2+\sigma_i^2}) \Big\}, \quad (6)$$

where π_i is a function of $\mathbf{g}$ and $\mathbf{h}$, and μ_i^{new} is a linear combination of $\tilde{b}_i$ and μ_i (Eq. 4). This means that $p(\tilde{v}_i|\mathbf{h},\mathbf{g})$ is a mixture of two Gaussians with different variances: one large ($\tilde{\sigma}_i^2$) and one much smaller, since $\tilde{\sigma} >> \sigma$. The mixing proportions are not fixed but rather *depend* on $\mathbf{g}$ and $\mathbf{h}$, which can learn the spatial structures (if any) of the corruptions.

The RoBM model is also a generalization of the common MRF framework used for image restoration and denoising [2, 16]. Setting $s_i = 1, \forall_i$, and $\frac{\gamma_i^2}{\sigma_i^2}$ to the noise variance of the data penalty, we recover an MRF model with the GRBM specifying its image prior instead of the usual local smoothness potentials. Whereas the parameters of the data penalty in standard MRFs are usually manually specified, the equivalent parameters in the RoBM, $s_i\gamma_i^2$, are actually random variables. We will show in section 2.2.2 that the distribution of $s_i\gamma_i^2$ can be learned from noisy data in an unsupervised fashion.

[1] There will still be a small penalty from the noise model.

2.2.1 Inference

Inference in the RoBM consists of finding the posterior distribution of the latent variables conditioned on the observed variables: $p(\mathbf{v},\mathbf{s},\mathbf{g},\mathbf{h}|\tilde{\mathbf{v}})$. This distribution is complicated but we can use the alternating Gibbs operator to sample from this posterior. Alternating Gibbs is much more efficient than standard Gibbs as we have two alternating conditional distributions which are easy to sample from.

Conditional 1: $p(\mathbf{v},\mathbf{s}|\mathbf{g},\mathbf{h},\tilde{\mathbf{v}})$

Conditional 2: $p(\mathbf{g},\mathbf{h}|\mathbf{v},\mathbf{s},\tilde{\mathbf{v}})$

Conditional 1: we can efficiently draw samples by first sampling $p(\mathbf{s}|\mathbf{g},\mathbf{h},\tilde{\mathbf{v}})$, then $p(\mathbf{v}|\mathbf{s},\mathbf{h},\tilde{\mathbf{v}})$, since:

$$p(\mathbf{v},\mathbf{s}|\mathbf{g},\mathbf{h},\tilde{\mathbf{v}}) = p(\mathbf{v}|\mathbf{s},\mathbf{h},\tilde{\mathbf{v}})p(\mathbf{s}|\mathbf{g},\mathbf{h},\tilde{\mathbf{v}}). \quad (7)$$

In addition, due to the form of Eq. 5, when given $\mathbf{g}$ and $\mathbf{h}$, the distribution over $\mathbf{v}$ and $\mathbf{s}$ is factorial:

$$\begin{aligned} p(\mathbf{v},\mathbf{s}|\mathbf{g},\mathbf{h},\tilde{\mathbf{v}}) &= \prod_i p(v_i,s_i|\mathbf{g},\mathbf{h},\tilde{\mathbf{v}}) \\ &= \prod_i p(v_i|s_i,\mathbf{g},\mathbf{h},\tilde{\mathbf{v}})p(s_i|\mathbf{g},\mathbf{h},\tilde{\mathbf{v}}). \quad (8) \end{aligned}$$

Moreover, it can be shown by integrating out v_i that

$$p(s_i|\mathbf{g},\mathbf{h},\tilde{\mathbf{v}}) = \frac{\alpha^{s_i}\beta^{1-s_i}}{\alpha+\beta} \quad (9)$$

$$\alpha = \hat{\sigma}_i \exp\Big\{(d_i + U_i\mathbf{g}) - \frac{1}{2}\frac{\gamma_i^2}{\sigma_i^2}\tilde{v}_i^2 + \frac{1}{2}\frac{\hat{\mu}_i^2}{\hat{\sigma}_i^2}\Big\} \quad (10)$$

$$\beta = \sigma_i \exp\Big\{\frac{1}{2}\frac{\mu_i^2}{\sigma_i^2}\Big\}, \quad (11)$$

where μ_i is defined by Eq. 4, $\hat{\mu} = \frac{\mu_i+\gamma_i^2\tilde{v}_i}{\gamma_i^2+1}$, and $\hat{\sigma}_i = \frac{\sigma_i}{\sqrt{\gamma_i^2+1}}$. Note that $d_i + U_i\mathbf{g}$ is the total input coming from the $\mathbf{g}$ layer, and $\boldsymbol{\mu}$ is the total input coming from the $\mathbf{h}$ layer. After sampling s_i, the conditional distribution over v_i is:

$$p(v_i|s_i,\mathbf{h},\tilde{\mathbf{v}}) \sim \mathcal{N}\Big(\frac{s_i\gamma_i^2}{s_i\gamma_i^2+1}\tilde{v}_i + \frac{\mu_i}{s_i\gamma_i^2+1}, \frac{\sigma_i^2}{s_i\gamma_i^2+1}\Big) \quad (12)$$

The above equation has a very intuitive interpretation. When $s_i = 0$, node i is corrupted, v_i is distributed according to $v_i \sim \mathcal{N}(\mu_i;\sigma_i^2)$, where μ_i is determined by the hidden nodes of the GRBM. However, when $s_i = 1$, node i

Algorithm 1 Inference in the RoBM: $p(\mathbf{v}, \mathbf{s}, \mathbf{h}, \mathbf{g}|\tilde{\mathbf{v}})$

Randomly initialize the layers of $\mathbf{h}$, $\mathbf{g}$.
for $t = 1 : NumberGibbsSteps$ **do**
  Sample from $p(\mathbf{s}|\mathbf{g}, \mathbf{h}, \tilde{\mathbf{v}})$, using Eq. 9.
  Sample from $p(\mathbf{v}|\mathbf{s}, \mathbf{h}, \tilde{\mathbf{v}})$, using Eq. 12.
  Sample from $p(\mathbf{g}, \mathbf{h}|\mathbf{v}, \mathbf{s}, \tilde{\mathbf{v}})$, using Eq. 13.
end for

is not corrupted, and its mean is a weighted average of μ_i and the observed input $\tilde{v}_i$. The weighting is determined by the parameter γ_i^2, which acts as the precision of the sensor noise. When it is large, v_i will be very similar to $\tilde{v}_i$. When it is small, v_i is allowed to be different from $\tilde{v}_i$ since its deviation can be explained by the observation noise.

Conditional 2: The 2nd conditional is efficient to compute as it can be factored into a product of the RBM and GRBM posteriors:

$$p(\mathbf{g}, \mathbf{h}|\mathbf{v}, \mathbf{s}, \tilde{\mathbf{v}}) = p(\mathbf{g}, \mathbf{h}|\mathbf{v}, \mathbf{s}) = p(\mathbf{h}|\mathbf{v})p(\mathbf{g}|\mathbf{s}), \quad (13)$$

where $p(\mathbf{h}|\mathbf{v}) = \prod_j p(h_j|\mathbf{v})$ (see Eq. 2). Similarly, we also have $p(\mathbf{g}|\mathbf{s}) = \prod_k p(g_k|\mathbf{s})$. The algorithm for performing posterior inference is shown in Alg. 1.

2.2.2 Learning

The parameters of the RoBM can be learned by maximizing the log-likelihood over the observed noisy images $\tilde{\mathbf{v}}$:

$$\hat{\theta} = \arg\max_{\theta} \ \log p(\tilde{\mathbf{v}}; \theta), \quad (14)$$

where θ is the collection of all parameters of the RoBM in Eq. 5. In an undirected graphical model, such as a Boltzmann Machine, maximum likelihood learning can be accomplished by gradient ascent, where gradients with respect to the parameters are given by the difference of two expectations:

$$\frac{\partial}{\partial\theta}\mathbb{E}[\log p(\tilde{\mathbf{v}}_n; \theta)] = \mathbb{E}_{model}\Big[\frac{\partial E_{RoBM}}{\partial\theta}\Big] - \mathbb{E}_{data}\Big[\frac{\partial E_{RoBM}}{\partial\theta}\Big]. \quad (15)$$

$\mathbb{E}_{model}[\cdot]$ denotes the expectation with respect to the distribution defined by the RoBM model (Eq. 5), while $\mathbb{E}_{data}[\cdot]$ denotes the empirical expectation with respect to the data distribution $p_{data}(\tilde{\mathbf{v}}, \mathbf{v}, \mathbf{s}, \mathbf{h}, \mathbf{g}) = p(\mathbf{v}, \mathbf{s}, \mathbf{h}, \mathbf{g}|\tilde{\mathbf{v}})p_{data}(\tilde{\mathbf{v}})$, where $p_{data}(\tilde{\mathbf{v}}) = \frac{1}{N}\sum_n \delta(\tilde{\mathbf{v}} - \tilde{\mathbf{v}}_n)$.

Exact maximum likelihood learning in this model is intractable, but efficient approximate learning can be done as follows. We first approximate $\mathbb{E}_{data}[\cdot]$ by sampling from the posterior $p(\mathbf{v}, \mathbf{s}, \mathbf{h}, \mathbf{g}|\tilde{\mathbf{v}})$ using a small number of alternating Gibbs updates (see Alg. 1). To approximate $\mathbb{E}_{model}[\cdot]$, we need to sample $\tilde{\mathbf{v}}$ as specified by the RoBM parameters. To sample from $\tilde{\mathbf{v}}$ given $\mathbf{v}$ and $\mathbf{s}$, we can sample each $\tilde{v}_i$ independently since $p(\tilde{\mathbf{v}}|\mathbf{v}, \mathbf{s}) = \prod_i p(\tilde{v}_i|\mathbf{v}, \mathbf{s})$. The conditional

Algorithm 2 Parameter Estimation for the RoBM

Pretrain the $\{\mathbf{v}, \mathbf{h}\}$ GRBM with "clean" data and initialize $\{\mathbf{W}, \mathbf{b}, \mathbf{c}\}$ of the RoBM with the pretrained parameters. Initialize other parameters randomly.
Initialize randomly the state of negative fantasy particles $\{\mathbf{v}^{fp}, \tilde{\mathbf{v}}^{fp}, \mathbf{s}^{fp}, \mathbf{g}^{fp}, \mathbf{h}^{fp}\}$ needed by PCD.
Initialize learning rate $\eta_0 \leftarrow 0.001$
for $m = 1 :$ number learning epochs **do**
  for $n = 1 :$ number of training cases **do**
    Use Alg. 1 to sample from $p(\mathbf{v}, \mathbf{s}, \mathbf{h}, \mathbf{g}|\tilde{\mathbf{v}}_n)$
    Calculate $\mathbb{E}_{data}\Big[\frac{\partial E_{RoBM}}{\partial\theta}\Big]$ using the samples of $\{\mathbf{v}, \mathbf{s}, \mathbf{h}, \mathbf{g}, \text{and } \tilde{\mathbf{v}}_n\}$.
    Use Alg. 1 sample from $p(\mathbf{v}^{fp}, \mathbf{s}^{fp}, \mathbf{h}^{fp}, \mathbf{g}^{fp}|\tilde{\mathbf{v}}^{fp})$
    Calculate $\mathbb{E}_{model}\Big[\frac{\partial E_{RoBM}}{\partial\theta}\Big]$ using the fantasy particles $\{\mathbf{v}^{fp}, \mathbf{s}^{fp}, \mathbf{h}^{fp}, \mathbf{g}^{fp}, \text{and } \tilde{\mathbf{v}}^{fp}\}$.
    Update: $\theta_{t+1} \leftarrow \theta_t + \frac{\partial \log p(\tilde{\mathbf{v}}_n)}{\partial\theta}$ (see Eq. 14).
  end for
  Decrease learning rate: $\eta_{t+1} = \eta_0/m$
end for

distribution over $\tilde{v}_i$ is a Gaussian distribution:

$$p(\tilde{v}_i|\mathbf{v}, \mathbf{s}) \sim \mathcal{N}\Big(\tilde{v}_i\Big|\alpha v_i + \beta\tilde{b}_i, \frac{\sigma_i^2\tilde{\sigma}_i^2}{\sigma_i^2 + s_i\gamma_i^2\tilde{\sigma}_i^2}\Big),$$
$$\alpha = \frac{s_i\gamma_i^2\tilde{\sigma}_i^2}{\sigma_i^2 + s_i\gamma_i^2\tilde{\sigma}_i^2}, \beta = \frac{\sigma_i^2}{\sigma_i^2 + s_i\gamma_i^2\tilde{\sigma}_i^2}.$$

The mean of this distribution is a linear combination of what the GRBM expects and what the noise term expects. In addition, the coefficients α and β depend on the random variable s_i. When $s_i = 0$, indicating that noise is present, $\tilde{v}_i$ is correctly sampled from the noise model with mean $\tilde{b}_i$ and variance $\tilde{\sigma}_i^2$.

During learning, we use a type of Stochastic Approximation of the Robbins-Monro type also known as Persistent Contrastive Divergence [23] to compute the model's expectation. Using PCD, we only need to run the Gibbs chain for a small number of iterations after each update of the parameters. With some mild conditions on the learning rates [28], we are guaranteed to converge to a locally optimal solution.

While it is possible to learn to maximize the objective function in Eq. 14 starting with random weights, it is much faster and easier if we first *pretrain* the parameters of the GRBM on "clean" data. It is not unreasonable for a model to have seen many noise-free examples of face images before learning on faces disguised with sunglasses. Learning is still unsupervised as *no* corresponding pairs of images of the same person, one with sunglasses and one without, are used during learning. The algorithm for RoBM learning is outlined in Alg. 2.

3. Experiments

We demonstrate the effectiveness of the RoBM on several standard face databases. Since the novelty of our model is in its ability to learn the structure and statistics from noisy data, we will first demonstrate it by using the Yale Face Database [1]. We will then show that denoising with the RoBM is significantly better than standard algorithm on the large Toronto Face Database [19]. Finally, we investigate the RoBM's recognition performance when test images contain noise or occlusions as in the Yale Database or contain disguise as in the AR Face Database [13].

3.1. Effects of Learning

We first demonstrate that RoBM's learning algorithm described in Sec. 2.2.2 can be successfully applied to learn directly from noisy data, without any knowledge of a clean image and its noisy version. We use the Yale Database for this experiment. The Yale Face Database contains 15 subjects with 11 images per subject. The face images are frontal but vary in illumination and expression. Following the standard protocol, we randomly select 8 images per subject as training and 3 for testing. We cropped images to the resolution of 32×32 and trained a GRBM model with visible nodes $\mathbf{v}$ and hidden nodes $\mathbf{h}$ on the "clean" faces. The training used Persistent Contrastive Divergence for a total of 50 epochs. We then initialized the RoBM's parameters $\{\mathbf{W}, \mathbf{b}, \mathbf{c}\}$ with the pretrained GRBM and applied the learning algorithm in Alg. 2 to learn the parameters of the RoBM model. In all of our experiments, $U_{ik}, e_k, \tilde{b}_i$ are initialized to 0.0, d_i to 4.0, γ_i to 20.0, and $\tilde{\sigma}_i^2$ is initialized to 1.0.

Fig. 3 shows the learning process of the RoBM. The columns represent the internal activation of the RoBM during learning from epoch 1 to epoch 50. The top row displays the training examples. The top panel shows an example that has a synthetically grid noise, while the bottom panel shows an example that has an occlusion by sunglasses. The second and third rows display the inferred faces $\mathbf{v}$ and the structure of the occluder/noise $\mathbf{s}$.

During the first learning epoch, the $\mathbf{U}$ matrix was initialized to zero. Therefore, no structure in $\mathbf{s}$ is modeled initially. This is confirmed by the fact that the inferred $\mathbf{s}$ are very noisy. As learning proceeds, we observe the trend that the actual shapes of the occluders are cleanly detected[2] and are modeled by the $\{\mathbf{s}, \mathbf{g}\}$ RBM. This demonstrates that we can in fact learn the noise structure in an unsupervised manner, when given a pretrained face density model.

To isolate the effect of having a model of the noise/occluder, we compare an RoBM model with hand-tuned parameters with an RoBM model trained on the noisy data. For the hand-tuned RoBM, we set its biases d_i such that the sigmoid of d_i would give the probability of each pixel being corrupted. Table 1 shows the PSNR of denoised Yale faces using an hand-tuned RoBM vs. an learned RoBM. For random noise, 40% of the pixels were corrupted by random noise with a standard deviation of 0.4. For block occlusion, 12×12 blocks were superimposed on a random part of the 32×32 faces.

For random noise, learning the structure of the noise does not add any value, thus similar results are expected. However, for block occlusions, structure learning helps denoising dramatically, resulting in an increase of 2 dB in the average denoised image.

[2]Some speckle will remain since we are viewing random samples from the posterior.

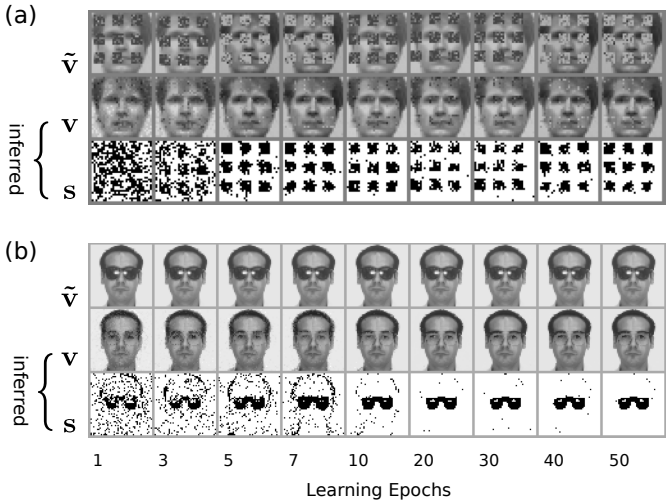

Figure 3. Internal states of the RoBM during learning: columns from left to right represent epochs 1 to 50. The first row is the training data $\tilde{\mathbf{v}}$, the second row is the inferred $\mathbf{v}$, and the third row is the inferred $\mathbf{s}$. 20 Gibbs iterations were run to sample from the posterior.

RoBMs Parameters	hand-tuned	learned
Random noise	30.0 ± 0.77	30.4 ± 0.88
Block occlusion	26.7 ± 0.85	28.6 ± 0.82

Table 1. Peak Signal to Noise Ratio (PSNR) in dB for denoising on Yale faces for a hand-tuned and learned RoBM. The numbers are averages over 40 trials $\pm$ the standard error of the mean.

3.2. Denoising

We next experimented on the large-scale Toronto Face Database (TFD) [19]. The TFD is a collection of (mostly) publicly available aligned face images. We used 60,000 training and 2,000 test 24×24 images. All test images are different from the training images by a Euclidean distance of at least 5.0. This eliminates cases where a test image is very similar to a training image, which is a possibility as the TFD faces were aggregated from a large collection of databases without separation by identity.

Figure 4. Difference between various denoising algorithms for block occlusion.

We first pretrained a GRBM model with 2,000 hidden nodes using Fast PCD [24] on the 60,000 training images for 500 epochs. The RoBM model was initialized exactly as described in the previous subsection. We then learned the joint model using Alg. 2. For block noise, we trained the RoBM model on data occluded by blocks at random positions. For random noise, we trained the RoBM model on data corrupted by random noise. After learning, we used 50 Gibbs iterations to sample from the posterior distribution. The denoised image of the RoBM model is taken to be the exponentially weighted average of the posterior samples with a weight of 0.9.

In all of our experiments, we compare performance of the RoBM model to the following four baseline models. Our first denoising algorithm, called RBM, consists of taking the pretrained GRBM model and initializing it with a noisy data. We run a few alternating Gibbs updates and take the exponentially weighted average as the denoised output. The second model, called PCA denoising algorithm, projects a noisy image onto a 75 dimensional subspace. The PCA reconstruction is then taken to be the denoised image. Our third algorithm performs Wiener filtering using MATLAB's *wiener2* function and a window size of 5. Our final baseline model finds the closest Euclidean nearest neighbor of the noisy test image in the training set.

Fig. 4 shows the denoising results for one face. The RoBM model performs significantly better than other methods. Since there is a dark occluder in the bottom left of the image, nearest neighbor found a different face with a shadow on the bottom left. While Wiener filtering works well for the Gaussian noise, it is not suitable for occlusions. PCA and RBM are unable to fully restore the occluded area, whereas RoBM is able to properly denoise due to its ability to gate off the occluded area and use its face prior to infer what is behind the occluder. We present similar qualitative results for random noise and occlusion in Fig. 5. Quantitatively, RoBM performs better than other models in terms of peak signal to noise ratio of the denoised results. Fig. 6 shows the results for both random noise and block occlusion.

We also investigated how sensitive our denoising results are to the hyper-parameter that specifies how many Gibbs iterations to run during inference. In Fig. 7, we plot the PSNR vs. the number of Gibbs iterations for both random noise and occlusion. From this plot, we see that 40 to 60 iterations tend to give the best *average* performance.

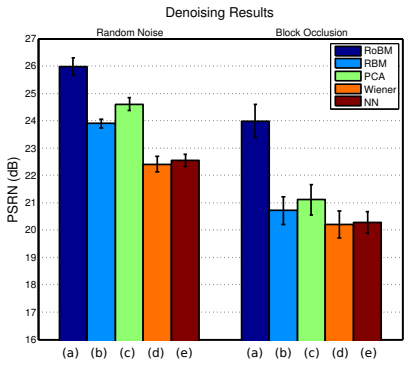

Figure 6. Quantitative denoising results. Methods: (a) RoBM, (b) RBM, (c) PCA, (d) Wiener, (e) Nearest Neighbor.

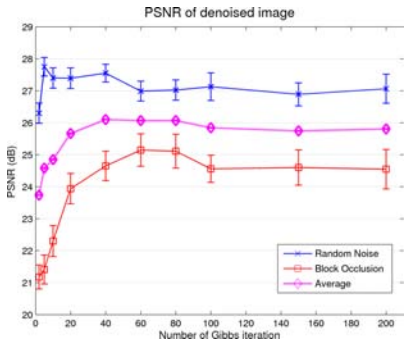

Figure 7. Denoising quality versus the number of Gibbs iterations used for sampling from the posterior during inference.

3.3. Recognition

In this section we test the ability of the RoBM to accurately recognize faces in the presence of noise and occlusion. We first add synthetic noise and occlusions to the faces in the Yale Database and plot classification accuracy as a function of the degree of noise/occlusion. We then test recognition performance with natural disguises (sunglasses and scarf) from the AR Face Database.

The classifier is a multi-class linear SVM trained on different feature representations of the faces. Recognition using the RoBM consists of first running 30 Gibbs iterations for denoising followed by classification using its hidden outputs before the sigmoid nonlinearity (Eq. 2). We provide comparisons to other benchmark models: pixels, LDA [12], Eigenfaces [25], and the standard GRBM. For the GRBM model, we first pretrain it and then run a few iterations of alternating Gibbs updates before classification.

Yale Face Database

As in Sec. 3.1, we used 8 images per subject for training and 3 for testing, and trained the RoBM model as specified in Sec. 3.1. During testing, for each noisy image, we ran 30 iterations of Gibbs sampling to arrive at a clean face. For classification, we feed the $\mathbf{h}$ layer activations (before the sigmoid nonlinearity) into the linear SVM. Fig. 8 shows

(a) Random Noise (b) Block Occlusions

Figure 5. Qualitative comparison of various denoising algorithms for two types of noise and occlusion. The first row has the original faces and the second row is corrupted with noise. Starting from the third row, we have denoising results from RoBM, RBM, PCA, Wiener filtering, and Nearest Neighbor, respectively.

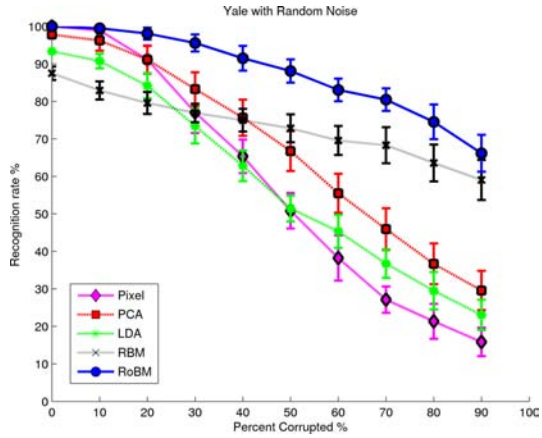

Figure 8. Recognition rates on the Yale Database as a function of the percentage of pixels corrupted by noise. Random noise with standard deviation of 0.5 were added to the corrupted pixels.

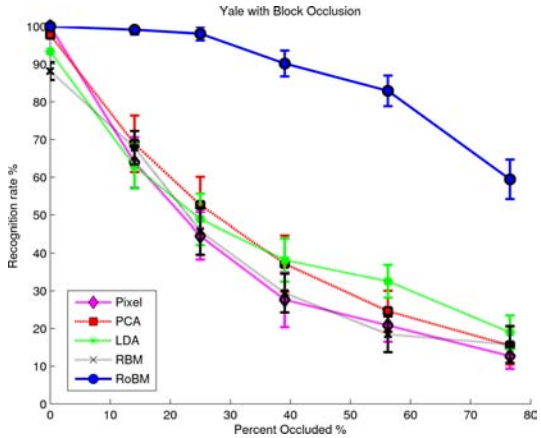

Figure 9. Recognition rates on the Yale Database as a function of the percentage of pixels occluded. Block occlusion were applied as in Fig. 5(b).

that RoBM performs better than other benchmark models, particularly when the amount of noise increases. The RBM method was run for 5 iterations of alternating Gibbs starting from the initial noisy image, where 5 iterations were chosen as it gave the best results on the test set. PCA used 50 eigenfaces and LDA was learned using the code provided by [12]. Fig. 9 displays the recognition accuracy as a function of the percentage of the image that is blocked.

AR Face Database

The AR database contains faces with real-life disguises using sunglasses and a scarf. We used a subset of 114 people each with 8 images for a total of 912 training images. For every person, there are two additional images with sunglasses and two with scarf occlusions, which we used as our test set. We first cropped and downsized the images to a resolution of 32×32 and pretrained a GRBM with 2000 hidden nodes. Initializing the RoBM model with weights from the pretrained GRBM, we learned one RoBM model on sunglasses and one RoBM model on the scarf images. After learning for 50 epochs, Fig. 10 displays the inferred "clean" face. Table 2 further shows that the RoBM model

Figure 10. Intermediate results during RoBM inference. The leftmost images are the test samples.

significantly outperforms all other models on the AR face recognition task.

Algorithms	Sunglasses	Scarf
RoBM	84.5 %	80.7 %
RBM	61.7 %	32.9 %
Eigenfaces	66.9 %	38.6 %
LDA	56.1 %	27.0 %
Pixel	51.3 %	17.5 %

Table 2. Recognition results on the AR Face Database.

4. Conclusions

We have described a novel model which allows Boltzmann Machines to be robust to noise and occlusions. By first training on noise-free images followed by unsupervised learning on noisy images, our model can learn the *structure* of the noise which allows it to perform much better on face denoising and recognition tasks.

References

[1] The yale face database., 2006. http://cvc.yale.edu/projects/yalefaces/yalefaces.html. 4

[2] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *PAMI*, PAMI-6(6):721–741, 1984. 3

[3] N. Heess, N. L. Roux, and J. Winn. Weakly supervised learning of foreground-background segmentation using masked RBMs. In *International Conference on Artificial Neural Networks (2011)*, July 19 2011. 1

[4] G. E. Hinton and R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313:504–507, 2006. 2

[5] P. J. Huber. *Robust Statistics*. Wiley series in probability and mathematical statistics. John Wiley & Sons, Inc., 1981. 3

[6] H. J. Jia and A. M. Martinez. Face recognition with occlusions in the training and testing sets. In *IEEE Int. Conf. on Automatic Face and Gesture Recognition*, 2008. 1

[7] A. Kannan, N. Jojic, and B. J. Frey. Generative model for layers of appearance and deformation. In *AISTATS, 2005*, 2005. 1

[8] A. Krizhevsky. Learning multiple layers of features from tiny images, 2009. 2

[9] Y. LeCun, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov. 1998. 1

[10] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Intl. Conf. on Machine Learning*, pages 609–616, 2009. 2

[11] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004. 1

[12] J. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos. Face recognition using kernel direct discriminant analysis algorithms. *IEEE-NN*, 14:117–126, Jan. 2003. 6, 7

[13] A. Martínez and R. Benavente. The ar face database, Jun 1998. 4

[14] A. Mohamed, G. Dahl, and G. Hinton. Acoustic modeling using deep belief networks. *IEEE Transactions on Audio, Speech, and Language Processing*, 2011. 1, 2

[15] M. Ranzato, J. Susskind, V. Mnih, and G. Hinton. On Deep Generative Models with Applications to Recognition. In *CVPR*, 2011. 1

[16] S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 860–867, 2005. 3

[17] N. L. Roux, N. Heess, J. Shotton, and J. M. Winn. Learning a generative model of images by factoring appearance and shape. *Neural Computation*, 23(3):593–650, 2011. 1

[18] P. Smolensky. Information processing in dynamical systems: Foundations of harmony theory. In D. E. Rumelhart and J. L. McClelland, editors, *Parallel Distributed Processing*, volume 1, chapter 6, pages 194–281. MIT Press, Cambridge, 1986. 2

[19] J. Susskind. The Toronto Face Database. Technical report, 2011. http://aclab.ca/users/josh/TFD.html. 4, 5

[20] M. Svensén and C. M. Bishop. Robust bayesian mixture modelling. *Neurocomputing*, 64:235–252, 2005. 3

[21] Y. Tang. Gated Boltzmann Machine for recognition under occlusion. In *NIPS Workshop on Transfer Learning by Learning Rich Generative Models*, 2010. 1

[22] G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler. Convolutional learning of spatio-temporal features. In *ECCV 2010*. Springer, 2010. 2

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

[24] T. Tieleman and G. E. Hinton. Using fast weights to improve persistent contrastive divergence. In *ICML*, volume 382, page 130. ACM, 2009. 6

[25] M. Turk and A. P. Pentland. Eigenfaces for recognition. *Journal Cognitive Neuroscience*, 3(1):71–96, 1991. 6

[26] C. K. I. Williams and M. K. Titsias. Greedy learning of multiple objects in images using robust statistics and factorial learning. *Neural Computation*, 16(5):1039–1062, May 2004. 1

[27] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 31(2):210–227, Feb. 2009. 1

[28] L. Younes. On the convergence of markovian stochastic algorithms with rapidly decreasing ergodicity rates. In *Stochastics and Stochastics Models*, pages 177–228, 1998. 4

[29] Z. Zhou, A. Wagner, H. Mobahi, J. Wright, and Y. Ma. Face recognition with contiguous occlusion using markov random fields. In *ICCV*, pages 1050–1057. IEEE, 2009. 1