Augmenting CRFs with Boltzmann Machine Shape Priors for Image Labeling

Andrew Kae*[1], Kihyuk Sohn*[2], Honglak Lee[2], Erik Learned-Miller[1]
[1] University of Massachusetts, Amherst, MA, USA, {akae,elm}@cs.umass.edu
[2] University of Michigan, Ann Arbor, MI, USA, {kihyuks,honglak}@umich.edu
* indicates equal contribution

Abstract

Conditional random fields (CRFs) provide powerful tools for building models to label image segments. They are particularly well-suited to modeling local interactions among adjacent regions (e.g., superpixels). However, CRFs are limited in dealing with complex, global (long-range) interactions between regions. Complementary to this, restricted Boltzmann machines (RBMs) can be used to model global shapes produced by segmentation models. In this work, we present a new model that uses the combined power of these two network types to build a state-of-the-art labeler. Although the CRF is a good baseline labeler, we show how an RBM can be added to the architecture to provide a global *shape bias that complements the local modeling provided by the CRF. We demonstrate its labeling performance for the parts of complex face images from the Labeled Faces in the Wild data set. This hybrid model produces results that are both quantitatively and qualitatively better than the CRF alone. In addition to high-quality labeling results, we demonstrate that the hidden units in the RBM portion of our model can be interpreted as face attributes that have been learned without any attribute-level supervision.*

1. Introduction

Segmentation and region labeling are core techniques for the critical mid-level vision tasks of grouping and organizing image regions into coherent parts. Segmentation refers to the grouping of image pixels into parts without applying labels to those parts, and region labeling assigns specific category names to those parts. While many segmentation and region labeling algorithms have been used in general object recognition and scene analysis, they have played a surprisingly small role in the challenging problems of face recognition.

Recently, Huang et al. [13] identified the potential role of region labeling in face recognition, noting that a variety of high-level features, such as pose, hair length, and gender can often be inferred (by people) from the labeling of a face image into hair, skin and background regions. They further showed that simple learning algorithms could be used to predict high-level features, or *attributes* [15, 24], such as pose, from the labeling.

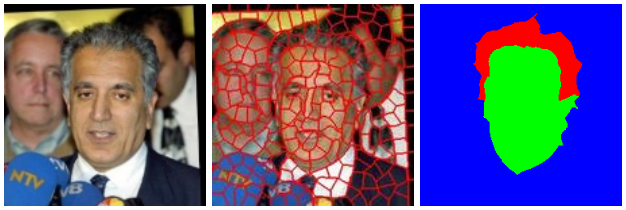


Figure 1. The left image shows a "funneled" or aligned LFW image. The center image shows the superpixel version of the image which is used as a basis for the labeling. The right image shows the ground truth labeling. Red represents hair, green represents skin, and the blue represents background.

In this work, we address the problem of labeling face regions with hair, skin, and background labels as an intermediate step in modeling face structure. In region labeling applications, the conditional random field (CRF) [16] is effective at modeling region boundaries. For example, the CRF can make a correct transition between the hair and background labels when there is a clear difference between those regions. However, when a person's hair color is similar to that of the background, the CRF may have difficulty deciding where to draw the boundary between the regions. In such cases, a global shape constraint can be used to filter out unrealistic label configurations.

It has been shown that restricted Boltzmann machines (RBMs) [28] and their extension to deeper architectures such as deep Boltzmann machines (DBMs) [25], can be used to build effective generative models of object shape. Specifically, the recently proposed shape Boltzmann machine (ShapeBM) [5] showed impressive performance in generating novel but realistic object shapes while capturing both local and global elements of shape.

Motivated by these examples, we propose the *GLOC* (GLObal and LOCal) model, a strong model for image labeling problems, that combines the best properties of the CRF (that enforces *local consistency* between adjacent nodes) and the RBM (that models *global shape prior* of the

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.263

object). The model balances three goals in seeking label assignments:

- The region labels should be consistent with the underlying image features.
- The region labels should respect image boundaries.
- The complete image labeling should be consistent with shape priors defined by the segmentation training data.

In our GLOC model, the first two objectives are achieved primarily by the CRF part, and the third objective is addressed by the RBM part. For each new image, our model uses mean-field inference to find a good balance between the CRF and RBM potentials in setting the image labels and hidden node values.

We evaluate our proposed model on a face labeling task using the Labeled Faces in the Wild (LFW) data set. As shown in Section 4, our model brings significant improvements in labeling accuracy over the baseline methods, such as the CRF and the conditional RBM. These gains in numerical accuracy have a significant visual impact on the resulting labeling, often fixing errors that are small but obvious to any observer. In addition, we show in Section 5 that the hidden units in the GLOC model can be interpreted as face attributes, such as whether an individual has long hair or a beard, or faces left or right. These attributes can be useful in retrieving face images with similar structure and properties.

We summarize our main contributions as follows:

- We propose the GLOC model, a strong model for face labeling tasks, that combines the CRF and the RBM to achieve both local and global consistency.
- We present efficient inference and training algorithms for our model.
- We achieve significant improvements over the state-of-the-art in face labeling accuracy on subsets of the LFW data set. Our model also produces qualitatively better labeling than the baseline CRF models.
- We demonstrate that our model learns face attributes automatically without attribute labels.

We will make the code [1] and part label data set [2] publicly available.

2. Prior Work

2.1. Face Segmentation and Labeling

Several authors have built systems for segmenting hair, skin, and other face parts [30, 29, 27, 19, 32, 13]. Because of the variety of hair styles, configurations, and amount of hair, the shape of a hair segmentation can be extremely variable. In our work, we treat facial hair as part of "hair" in general, hoping to develop hidden units corresponding to beards, sideburns, mustaches, and other hair parts, which further increases the complexity of the hair segments. Furthermore, we include skin of the neck as part of the "skin" segmentation when it is visible, which is different from other labeling regimes. For example, Wang et al. [29] limit the skin region to the face and include regions covered by beards, hats, and glasses as being skin, which simplifies their labeling problem.

Yacoob et al. [32] build a hair color model and then adopt a region growing algorithm to modify the hair region. This method has difficulty when the hair color changes significantly from one region to another, especially for dark hair, and the work was targeted at images with controlled backgrounds. Lee et al. [19] used a mixture model to learn six distinct hair styles, and other mixture models to learn color distributions for hair, skin, and background.

Huang et al. [13] used a standard CRF trained on images from LFW to build a hair, skin, and background labeler. We have implemented their model as a baseline and report the performance. Scheffler et al. [27] learn color models for hair, skin, background and clothing. They also learn a spatial prior for each label. They combine this information with a Markov random field that enforces local label consistency.

Wang et al. [29] used a compositional exemplar-based model, focusing mostly on the problem of hair segmentation. Following their earlier work, Wang et al. [30] proposed a model that regularizes the output of a segmentation using parts. In addition, their model builds a statistical model of *where* each part is used in the image and the co-occurrence probabilities between parts. Using these co-occurrences, they build a tree-structured model over parts to constrain the final segmentations. To our knowledge, this is the best-performing algorithm for hair, skin, and background labeling to date. In Section 4, we report the results on two sets of labeled data showing improvements over these best previous results.

2.2. Object Shape Modeling

There are several related works on using RBMs (or their deeper extensions) for shape modeling. He et al. [7] proposed multiscale CRFs to model both local and global label features using RBMs. Specifically, they used multiple RBMs at different scales to model the regional or global label fields (layers) separately, and combined those conditional distributions multiplicatively. Recent work by Eslami et al. [5] introduced the Shape Boltzmann machine (ShapeBM), a two-layer DBM with local connectivity in the first layer for local consistency and generalization (by weight sharing), and full connectivity in the second layer for modeling global shapes, as a strong model for object shapes. Subsequently, Eslami and Williams [6] proposed a generative model by combining the ShapeBM with an appearance model for parts-based object segmentation. Our model is similar at a high-level to these models in that we use RBMs for object shape modeling to solve image labeling problems. However, there are significant technical differences that distinguish our model from others. First, our model has an edge potential that enforces local consistency between adjacent superpixel labels. Second, we define our model on the superpixel graph using a virtual pooling technique, which is computationally much more efficient. Third, our model is discriminative and can use richer image features than [6] which used a simple pixel-level appearance model (based on RGB pixel values). Finally, we propose a model combined with an RBM to act as a shape prior, which makes the training much easier while showing significant improvement over the baseline models in face labeling tasks. See 3.2.4 for more discussions.

3. Algorithms

In this section, we briefly describe the CRF and RBM, followed by our proposed GLOC model. We present the models in the context of multi-class labeling.

Notation An image I is pre-segmented into $S^{(I)}$ superpixels, where $S^{(I)}$ can vary over different images. We denote $\mathcal{V}^{(I)} = \{1, \cdots, S^{(I)}\}$ as a set of superpixel nodes, and $\mathcal{E}^{(I)}$ as a set of edges connecting adjacent superpixels. We denote $\mathcal{X}^{(I)} = \{\mathcal{X}_{\mathcal{V}}^{(I)}, \mathcal{X}_{\mathcal{E}}^{(I)}\}$, where $\mathcal{X}_{\mathcal{V}}^{(I)}$ is a set of node features $\{\mathbf{x}_s^{\text{node}} \in \mathbb{R}^{D_n}, s \in \mathcal{V}\}$ and $\mathcal{X}_{\mathcal{E}}^{(I)}$ is a set of edge features $\{\mathbf{x}_{ij}^{\text{edge}} \in \mathbb{R}^{D_e}, (i,j) \in \mathcal{E}\}$. The set of label nodes are defined as $\mathcal{Y}^{(I)} = \{\mathbf{y}_s \in \{0,1\}^L, s \in \mathcal{V} : \sum_{l=1}^{L} y_{sl} = 1\}$. Here, D_n and D_e denote the dimensions of the node and edge features, respectively, and L denotes the number of categories for the labeling task. We frequently omit the superscripts "I", "node", or "edge" for clarity, but the meaning should be clear from the context.

3.1. Preliminaries

3.1.1 Conditional Random Fields

The conditional random field [16] is a powerful model for structured output prediction (such as sequence prediction, text parsing, and image segmentation), and has been widely used in computer vision [8, 3, 4, 7]. The conditional distribution and the energy function can be defined as follows:

$$P_{\text{crf}}(\mathcal{Y}|\mathcal{X}) \propto \exp(-E_{\text{crf}}(\mathcal{Y},\mathcal{X})), \quad (1)$$

$$E_{\text{crf}}(\mathcal{Y},\mathcal{X}) = E_{\text{node}}(\mathcal{Y},\mathcal{X}_{\mathcal{V}}) + E_{\text{edge}}(\mathcal{Y},\mathcal{X}_{\mathcal{E}}), \quad (2)$$

$$E_{\text{node}}(\mathcal{Y},\mathcal{X}_{\mathcal{V}}) = -\sum_{s\in\mathcal{V}}\sum_{l=1}^{L}\sum_{d=1}^{D_n} y_{sl}\Gamma_{ld}x_{sd},$$

$$E_{\text{edge}}(\mathcal{Y},\mathcal{X}_{\mathcal{E}}) = -\sum_{(i,j)\in\mathcal{E}}\sum_{l,l'=1}^{L}\sum_{e=1}^{D_e} y_{il}y_{jl'}\Psi_{ll'e}x_{ije},$$

where $\Psi \in \mathbb{R}^{L\times L\times D_e}$ is a 3D tensor for the edge weights, and $\Gamma \in \mathbb{R}^{L\times D_n}$ are the node weights. The model parameters $\{\Gamma, \Psi\}$ are trained to maximize the conditional log-likelihood of the training data $\{\mathcal{Y}^{(m)}, \mathcal{X}^{(m)}\}_{m=1}^{M}$,

$$\max_{\Gamma,\Psi}\sum_{m=1}^{M}\log P_{\text{crf}}(\mathcal{Y}^{(m)}|\mathcal{X}^{(m)}).$$

We can use loopy belief propagation (LBP) [23] or mean-field approximation [26] for inference in conjunction with standard optimization methods such as LBFGS.[1]

3.1.2 Restricted Boltzmann Machines

The restricted Boltzmann machine [28] is a bipartite, undirected graphical model composed of visible and hidden layers. In our context, we assume R^2 multinomial visible units $\mathbf{y}_r \in \{0,1\}^L$ and K binary hidden units $h_k \in \{0,1\}$. The joint distribution can be defined as follows:

$$P_{\text{rbm}}(\mathcal{Y},\mathbf{h}) \propto \exp(-E_{\text{rbm}}(\mathcal{Y},\mathbf{h})), \quad (3)$$

$$E_{\text{rbm}}(\mathcal{Y},\mathbf{h}) = -\sum_{r=1}^{R^2}\sum_{l=1}^{L}\sum_{k=1}^{K} y_{rl}W_{rlk}h_k - \sum_{k=1}^{K} b_k h_k - \sum_{r=1}^{R^2}\sum_{l=1}^{L} c_{rl}y_{rl}, \quad (4)$$

where $\mathbf{W} \in \mathbb{R}^{R^2\times L\times K}$ is a 3D tensor specifying the connection weights between visible and hidden units, b_k is the hidden bias, and c_{rl} is the visible bias. The parameters $\Theta = \{\mathbf{W}, \mathbf{b}, \mathbf{C}\}$ are trained to maximize the log-likelihood of the training data $\{\mathcal{Y}^{(m)}\}_{m=1}^{M}$,

$$\max_{\Theta}\sum_{m=1}^{M}\log\left(\sum_{\mathbf{h}} P_{\text{rbm}}(\mathcal{Y}^{(m)},\mathbf{h})\right).$$

We train the model parameters using stochastic gradient descent. Although the exact gradient is intractable to compute, we can approximate it using contrastive divergence [9].

3.2. The GLOC Model

To build a strong model for image labeling, both local consistency (adjacent nodes are likely to have similar labels) and global consistency (the overall shape of the object should look realistic) are desirable. On one hand, the CRF is powerful in modeling local consistency via edge potentials. On the other hand, the RBM is good at capturing global shape structure through the hidden units. We combine these two ideas in the GLOC model, which incorporates both local consistency (via CRF-like potentials) and global consistency (via RBM-like potentials). Specifically, we describe the conditional likelihood of labels set $\mathcal{Y}$ given the superpixel features $\mathcal{X}$ as follows:

$$P_{\text{gloc}}(\mathcal{Y}|\mathcal{X}) \propto \sum_{\mathbf{h}} \exp\left(-E_{\text{gloc}}(\mathcal{Y},\mathcal{X},\mathbf{h})\right), \quad (5)$$

$$E_{\text{gloc}}(\mathcal{Y},\mathcal{X},\mathbf{h}) = E_{\text{crf}}(\mathcal{Y},\mathcal{X}) + E_{\text{rbm}}(\mathcal{Y},\mathbf{h}). \quad (6)$$

[1] We used LBFGS in minFunc by Mark Schmidt: http://www.di.ens.fr/~mschmidt/Software/minFunc.html

As described above, the energy function is written as a combination of CRF and RBM energy functions. However, due to the varying number of superpixels for different images, the RBM energy function in Equation (4) requires nontrivial modifications. In other words, we cannot simply connect label (visible) nodes defined over superpixels to hidden nodes as in Equation (4) because 1) the RBM is defined on a fixed number of visible nodes and 2) the number of superpixels and their underlying graph structure can vary across images.

3.2.1 Virtual Pooling Layer

To resolve this issue, we introduce a *virtual, fixed-sized* pooling layer between the label and the hidden layers, where we map each superpixel label node into the *virtual* visible nodes of the $R \times R$ square grid. This is shown in Figure 2, where the top two layers can be thought of as an RBM with the visible nodes $\bar{\mathbf{y}}_r$ representing a surrogate (i.e., pooling) for the labels $\mathbf{y}_s$ that overlap with the grid bin r. Specifically, we define the energy function between the label nodes and the hidden nodes for an image I as follows:

$$E_{\text{rbm}}\left(\mathcal{Y}, \mathbf{h}\right) = -\sum_{r=1}^{R^2}\sum_{l=1}^{L}\sum_{k=1}^{K} \bar{y}_{rl} W_{rlk} h_k - \sum_{k=1}^{K} b_k h_k - \sum_{r=1}^{R^2}\sum_{l=1}^{L} c_{rl}\bar{y}_{rl}. \qquad (7)$$

Here, the virtual visible nodes $\bar{y}_{rl} = \sum_{s=1}^{S} p_{rs} y_{sl}$ are deterministically mapped from the superpixel label nodes using the projection matrix $\{p_{rs}\}$ that determines the contribution of label nodes to each node of the grid. The projection matrix is defined as follows:[2]

$$p_{rs} = \frac{Area(Region(s) \cap Region(r))}{Area(Region(r))},$$

where $Region(s)$ and $Region(r)$ denote sets of pixels corresponding to superpixel s and grid r, respectively. Due to the deterministic connection, the pooling layer is actually a *virtual* layer that only exists to map between the superpixel nodes and the hidden nodes. We can also view our GLOC model as having a set of grid-structured nodes that performs average pooling over the adjacent superpixel nodes.

3.2.2 Spatial CRF

As an additional baseline, we describe a modification to the CRF presented in Section 3.1.1. In some cases, even after conditioning on class, feature likelihoods may depend on position. For example, knowing that hair rests on the shoulders makes it less likely to be gray. This intuition is behind our Spatial CRF model.

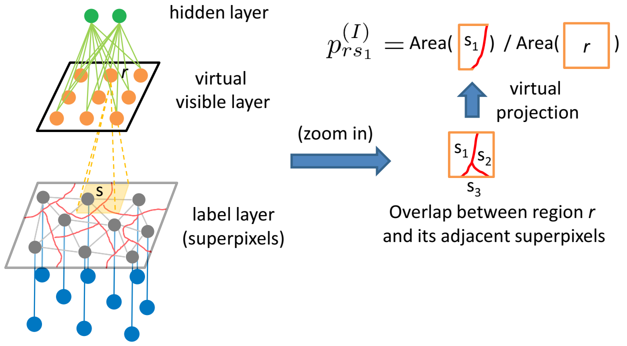

hidden layer
virtual visible layer
r
s
label layer (superpixels)
(zoom in)
$p_{rs_1}^{(I)}$ = Area(s1) / Area(r)
virtual projection
s1
s2
s3
Overlap between region r and its adjacent superpixels

Figure 2. The GLOC model. The top two layers can be thought of as an RBM with the (virtual) visible nodes $\bar{\mathbf{y}}_r$ and the hidden nodes. To define the RBM over a fixed-size visible node grid, we use an image-specific "projection matrix" $\{p_{rs}^{(I)}\}$ that transfers (top-down and bottom-up) information between the label layer and the virtual grid of the RBM's visible layer. See text for details.

Specifically, when the object in the image is aligned, we can learn a spatially dependent set of weights that are specific to a cell in an $N \times N$ grid. (Note that this grid can be a different size than the $R \times R$ grid used by the RBM.) We learn a separate set of node weights for each cell in a grid, but the edge weights are kept globally stationary.

Using a similar projection technique to that described in Section 3.2.1, we define the node energy function as

$$E_{\text{node}}\left(\mathcal{Y}, \mathcal{X}_{\mathcal{V}}\right) = -\sum_{s\in\mathcal{V}}\sum_{l=1}^{L} y_{sl} \sum_{n=1}^{N^2} p_{sn} \sum_{d=1}^{D_n} \Gamma_{ndl} x_{sd}, \qquad (8)$$

where $\Gamma \in \mathbb{R}^{N^2 \times D \times L}$ is a 3D tensor specifying the connection weights between the superpixel node features and labels at each spatial location. In this energy function, we define a different projection matrix $\{p_{sn}\}$ which specifies the mapping from the $N \times N$ virtual grid to superpixel label nodes.[3]

3.2.3 Inference and Learning

Inference Since the joint inference of superpixel labels and the hidden nodes is intractable, we resort to the mean-field approximation. Specifically, we find a fully factorized distribution $Q(\mathcal{Y}, \mathbf{h}; \mu, \gamma) = \prod_{s\in\mathcal{V}} Q(\mathbf{y}_s) \prod_{k=1}^{K} Q(h_k)$, with $Q(\mathbf{y}_s = l) \triangleq \mu_{sl}$ and $Q(h_k = 1) \triangleq \gamma_k$, that minimizes $\text{KL}\left(Q(\mathcal{Y}, \mathbf{h}; \mu, \gamma) \| P(\mathcal{Y}, \mathbf{h} | \mathcal{X})\right)$. We describe the mean-field inference steps in Algorithm 1.

Learning In principle, we can train the model parameters $\{\mathbf{W}, \mathbf{b}, \mathbf{C}, \Gamma, \Psi\}$ simultaneously to maximize the conditional log-likelihood. In practice, however, it is beneficial

[2] The projection matrix $\{p_{rs}\}$ is a sparse, non-negative matrix of dimension $R^2 \times S$. Note that the projection matrix is specific to each image since it depends on the structure of the superpixel graph.

[3] Note that the projection matrices used in the RBM and spatial CRF are different in that $\{p_{rs}\}$ used in the RBM describes a projection from superpixel to grid ($\sum_{s=1}^{S} p_{rs} = 1$), whereas $\{p_{sn}\}$ used in the spatial CRF describes a mapping from a grid to superpixel ($\sum_{n=1}^{N^2} p_{sn} = 1$).

Algorithm 1 Mean-Field Inference

```
Initialize $\boldsymbol{\mu}^{(0)}$ and $\boldsymbol{\gamma}^{(0)}$ as follows:
    $\mu_{sl}^{(0)} = \frac{\exp\left(f_{sl}^{\text{node}}\right)}{\sum_{l'} \exp\left(f_{sl'}^{\text{node}}\right)}$
    $\gamma_k^{(0)} = sigmoid\left(\sum_{r,l}\left(\sum_s p_{rs}\mu_{sl}^{(0)}\right) W_{rlk} + b_k\right)$
  where
    $f_{sl}^{\text{node}}\left(\mathcal{X}_\mathcal{V}, \{p_{sn}\}, \Gamma\right) = \sum_{n,d} p_{sn} x_{sd} \Gamma_{ndl}$
for t=0:$maxiter$ (or until convergence) do
    update $\boldsymbol{\mu}^{(t+1)}$ as follows: $\mu_{sl}^{(t+1)} =$
        $\frac{\exp\left(f_{sl}^{\text{node}} + f_{sl}^{\text{edge}}\left(\boldsymbol{\mu}^{(t)}\right) + f_{sl}^{\text{rbm}}\left(\boldsymbol{\gamma}^{(t)}\right)\right)}{\sum_{l'} \exp\left(f_{sl'}^{\text{node}} + f_{sl'}^{\text{edge}}\left(\boldsymbol{\mu}^{(t)}\right) + f_{sl'}^{\text{rbm}}\left(\boldsymbol{\gamma}^{(t)}\right)\right)}$
      where
        $f_{sl}^{\text{edge}}\left(\boldsymbol{\mu}; \mathcal{X}_\mathcal{E}, \mathcal{E}, \Psi\right) = \sum_{j:(s,j)\in\mathcal{E}} \sum_{l',e} \mu_{jl'} \Psi_{ll'e} x_{sje}$
        $f_{sl}^{\text{rbm}}\left(\boldsymbol{\gamma}; \{p_{rs}\}, \mathbf{W}, \mathbf{C}\right) = \sum_{r,k} p_{rs}\left(W_{rlk}\gamma_k + c_{rl}\right)$
    update $\boldsymbol{\gamma}^{(t+1)}$ as follows:
        $\gamma_k^{(t+1)} = sigmoid\left(\sum_{r,l}\left(\sum_s p_{rs}\mu_{sl}^{(t+1)}\right) W_{rlk} + b_k\right)$
end for
```

to provide a proper initialization (or *pretrain*) to those parameters. We provide an overview of the training procedure in Algorithm 2.

First, we adapted the pretraining method of deep Boltzmann machines (DBM) [25] to train the conditional RBM (CRBM).[4] Specifically, we pretrain the model parameters $\{\mathbf{W}, \mathbf{b}, \mathbf{C}\}$ of the CRBM as if it is a top layer of the DBM to avoid double-counting when combined with the edge potential in the GLOC model. Second, the CRBM and the GLOC models can be trained to either maximize the conditional log-likelihood using contrastive divergence (CD) or *minimize generalized perceptron loss* [18] using CD-PercLoss [22]. In fact, Mnih et al. [22] suggested that CD-PercLoss would be a better choice for structured output prediction problems since it directly penalizes the model for wrong predictions during training. We empirically observed that CD-PercLoss performed slightly better than CD.

Algorithm 2 Training GLOC model

```
Pretrain $\{\Gamma, \Psi\}$ to maximize the conditional log-likelihood of the *spatial CRF* model (See Equations (1), (2), and (8)).
Pretrain $\Theta = \{\mathbf{W}, \mathbf{b}, \mathbf{C}\}$ to maximize the conditional log-likelihood $\log \sum_\mathbf{h} P_{\text{crbm}}(\mathcal{Y}, \mathbf{h}|\mathcal{X}_\mathcal{V})$ of the *conditional RBM* model which is defined as:
    $P_{\text{crbm}}(\mathcal{Y}, \mathbf{h}|\mathcal{X}_\mathcal{V}) \propto \exp\left(-E_{\text{node}}(\mathcal{Y}, \mathcal{X}_\mathcal{V}; \Gamma) - E_{\text{rbm}}(\mathcal{Y}, \mathbf{h}; \Theta)\right)$
Train $\{\mathbf{W}, \mathbf{b}, \mathbf{C}, \Gamma, \Psi\}$ to maximize the conditional log-likelihood of the *GLOC* model (See Equation (5)).
```

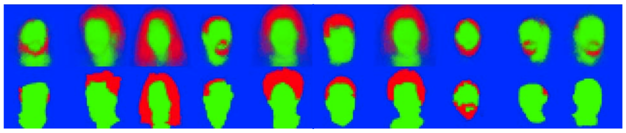


Figure 3. Generated samples from the RBM (first row) and the closest matching examples in the training set (second row). The RBM can generate novel, realistic examples by combining hair, beard and mustache shapes along with diverse face shapes.

3.2.4 Discussion

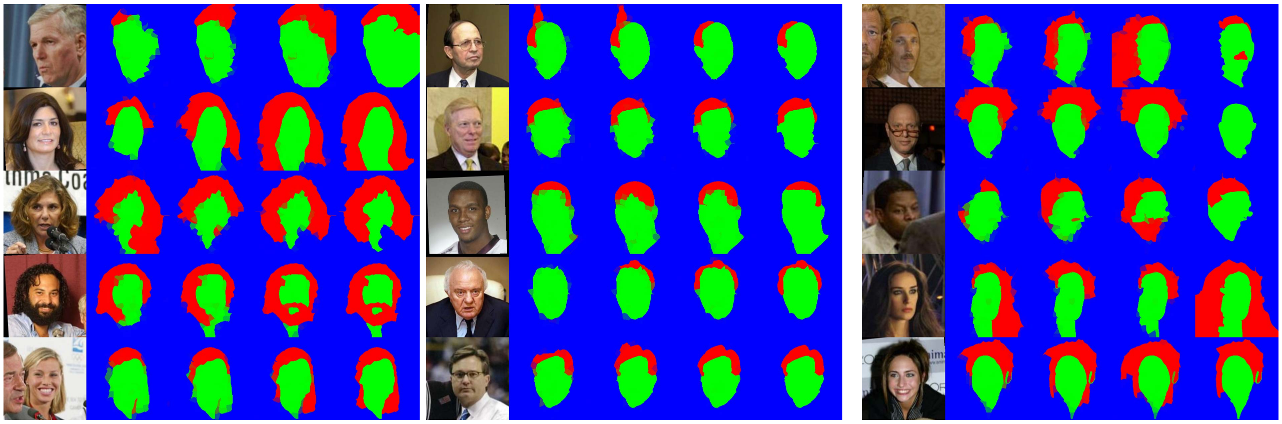
In many cases, it is advantageous to learn generative models with deep architectures. In particular, Eslami et al. [5] suggest that the ShapeBM, a special instance of the DBM, can be a better generative model than the RBM when they are only given several hundred training examples. However, when given sufficient training data (e.g., a few thousand), we found that the RBM can still learn a global shape prior with good generalization performance. In Figure 3, we show both generated samples from an RBM and their closest training examples.[5] The generated samples are diverse and are clearly different from their most similar examples in the training set. This suggests that our model is learning an interesting decomposition of the shape distributions for faces. Furthermore, RBMs are easier to train than DBMs in general, which motivates the use of RBMs in our model. In principle, however, we can also use such deep architectures in our GLOC model as a rich global shape prior without much modification to inference and learning.

4. Experiments

We evaluated our proposed model on a task to label face images from the LFW data set [14] as hair, skin, and background. We used the "funneled" version of LFW, in which images have been coarsely aligned using a congealing-style joint alignment approach [10]. Although some better automatic alignments of these images exist, such as the LFW-a data set [31], LFW-a does not contain color information, which is important for our application.

[4] Note that our CRBM is different from the one defined in [22] in that 1) our model has no connection between the conditioning nodes $\mathcal{X}$ and the hidden nodes, and 2) our model uses a projection (e.g., virtual pooling) matrix to deal with the varying number of label nodes over the images.

[5] We compute the L_2 distance between the generated samples and the examples in the training set.



(a) Successful examples

(b) Failure examples

Figure 4. Sample segmentation results on images from the LFW data set. The images contain extremely challenging scenarios such as multiple distractor faces, occlusions, strong highlights, and pose variation. The left of Figure 4(a) shows images in which the GLOC model made relatively large improvements to the baseline. The right of Figure 4(a) shows more subtle changes made by our model. The results in Figure 4(b) show typical failure cases. The columns correspond to 1) original image which has been aligned to a canonical position using funneling [10], 2) CRF, 3) spatial CRF, 4) GLOC and 5) ground truth labeling. Note that the CRBM model results are not shown here.

The LFW website provides the segmentation of each image into superpixels, which are small, relatively uniform pixel groupings.[6] We provide ground truth for a set of 2927 LFW images by labeling each superpixel as either hair, skin, or background [2]. While some superpixels may contain pixels from more than one region, most superpixels are generally "pure" hair, skin, or background.

There are several reasons why we used superpixel labeling instead of pixel labeling for this problem. First, the superpixel representation is computationally much more efficient. The number of nodes would be too large for pixel labeling since the LFW images are of size 250 × 250. However, each image can be segmented into 200-250 superpixels, resulting in the same number of nodes in the CRF, and this allowed us to do tractable inference using LBP or mean-field. In addition, superpixels can help smooth features such as color. For example, if the superpixel is mostly black but contains a few blue pixels, the blue pixels will be smoothed out from the feature vector, which can simplify inference.

We adopted the same set of features as in Huang et al. [13]. For each superpixel we used the following node features:

- Color: Normalized histogram over 64 bins generated by running K-means over pixels in LAB space.
- Texture: Normalized histogram over 64 textons which are generated according to [20].
- Position: Normalized histogram of the proportion of a superpixel that falls within each of the 8 × 8 grid elements on the image.[7]

Method	Accuracy (SP)	Error Reduction
CRF	93.23 %	–
Spatial CRF	93.95 %	10.64%
CRBM	94.10 %	12.85 %
GLOC	**94.95** %	**25.41** %

Table 1. Labeling accuracies for each model. We report the superpixel-wise labeling accuracy in the second column, and the error reduction over the CRF in the third column.

The following edge features were computed between adjacent superpixels:

- Sum of PB [21] values along the border.
- Euclidean distance between mean color histograms.
- Chi-squared distance between texture histograms as computed in [13].

We evaluated the labeling performance of four different models: a standard CRF, the spatial CRF, the CRBM, and our GLOC model. We provide the summary results in Table 1. We divided the labeled examples into training, validation, and testing sets that contain 1500, 500, and 927 examples, respectively. We trained our model using batch gradient descent and selected the model hyperparameters that performed best on the validation set. After cross-validation, we set K=400, R=24, and N=16. Finally, we evaluated that model on the test set. On a multicore AMD Opteron, average inference time per example was 0.254 seconds for the GLOC model and 0.063 seconds for the spatial CRF.

As shown in Table 1, the GLOC model substantially improves the superpixel labeling accuracy over the baseline CRF model as well as the spatial CRF and CRBM models. While absolute accuracy improvements (necessarily) become small as accuracy approaches 95%, the reduction

[6] Available at http://vis-www.cs.umass.edu/lfw/lfw_funneled_superpixels_fine.tgz.

[7] Note that the position feature is only used in the CRF.

in errors are substantial.

Furthermore, there are significant qualitative differences in many cases, as we illustrate in Figure 4(a). The samples on the left show significant improvement over the spatial CRF, and the ones on the right show more subtle changes made by the GLOC model. Here, we represent the confidence of the guess (posterior) by color intensity. The confident guess appears as a strong red, green, or blue color, and a less confident guess appears as a lighter mixture of colors. As we can see, the global shape prior of the GLOC model helps "clean up" the guess made by the spatial CRF in many cases, resulting in a more confident prediction.

In many cases, the RBM prior encourages a more realistic segmentation by either "filling in" or removing parts of the hair or face shape. For example, the woman in the second row on the left set recovers the left side of her hair and gets a more recognizable hair shape under our model. Also, the man in the first row on the right set gets a more realistic looking hair shape by removing the small (incorrect) hair shape on top of his head. This effect may be due to the top-down global prior in our GLOC model, whereas simpler models such as the spatial CRF do not have this information. In addition, there were cases (such as the woman in the fifth row of the left set) where an additional face in close proximity to the centered face may confuse the model. In this case, the CRF and spatial CRF models make mistakes, but since the GLOC model has a strong shape model, it was able to find a more recognizable segmentation of the foreground face.

On the other hand, the GLOC model sometimes makes errors. We show typical failure examples in Figure 4(b). As we can see, the model made significant errors in their hair regions. Specifically, in the first row, the hair of a nearby face is similar in color to the hair of the foreground face as well as the background, and our model incorrectly guesses more hair by emphasizing the hair shape prior, perhaps too strongly. In addition, there are cases in which occlusions cause problems, such as the third row. However, we point out that the occlusions are frequently handled correctly by our model (e.g., the microphone in the third row of the left set in Figure 4(a)).

4.1. Comparison to Prior Work

We also evaluated our model on the data set used in [30]. This data set contains 1046 LFW (unfunneled) images whose pixels are manually labeled for 4 regions (Hair, Skin, Background, and Clothing). Following their evaluation setup, we randomly split the data in half and used one half for training data and the other half for testing. We repeated this procedure five times, and report the average pixel accuracy as a final result.

We first generated the superpixels and features for each image, then ran our GLOC model to get label guesses for each superpixel, and finally mapped back to pixels for evaluation (it was necessary to map to pixels at the end because the ground truth is provided in pixels). We noticed that even with a perfect superpixel labeling, this mapping already incurs approximately 3% labeling error. However, our approach was sufficient to obtain a good pixel-wise accuracy of 90.7% (91.7% superpixel-wise accuracy), which improves by 0.7% upon their best reported result of 90.0%. The ground truth for a superpixel is a normalized histogram of the pixel labels in the superpixel.

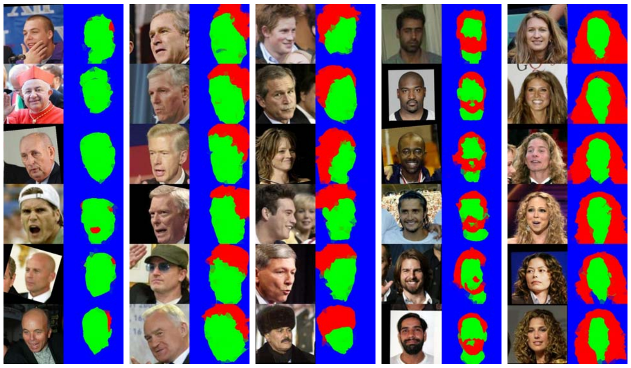


Figure 5. This figure shows some of the latent structure automatically learned by the GLOC model. In each column, we retrieve the images from LFW (except images used in training and validation) with the highest activations for each of 5 hidden units, and provide their segmentation results. The attributes from left to right can be interpreted as "no hair showing", "looking left", "looking right", "beard/occluded chin", "big hair". Although the retrieved matches are not perfect, they clearly have semantic, high-level content.

5. Attributes and Retrieval

While the labeling accuracy (as shown in Section 4) is a direct way of measuring progress, we have an additional goal in our work: to build models that capture the natural statistical structure in faces. It is not an accident that human languages have words for beards, baldness, and other salient high-level attributes of human face appearance. These attributes represent coherent and repeated structure across the faces we see everyday. Furthermore, these attributes are powerful cues for recognition, as demonstrated by Kumar et al. [15].

One of the most exciting aspects of RBMs and their deeper extensions are that these models can learn latent structure automatically. Recent work has shown that unsupervised learning models can learn meaningful structure without being explicitly trained to do so (e.g., [17, 11, 12]).

In our experiments, we ran our GLOC model on all LFW images other than those used in training and validation, and sorted them based on each hidden unit activation. Each of the five columns in Figure 5 shows a set of retrieved images and their guessed labelings for a particular hidden unit. In many cases, the retrieved results for the hidden units form meaningful clusters. These units seem highly correlated with "lack of hair", "looking left", "looking right", "beard or occluded chin", and "big hair". Thus, the learned hidden units may be useful as attribute representations for faces.

6. Conclusion

Face segmentation and labeling is challenging due to the diversity of hair styles, head poses, clothing, occlusions, and other phenomena that are difficult to model, especially in a database like LFW. Our GLOC model combines the CRF and the RBM to model both local and global structure in face segmentations. Our model has consistently reduced the error in face labeling over previous models which lack global shape priors. In addition, we have shown that the hidden units in our model can be interpreted as face attributes, which were learned without any attribute-level supervision.

Acknowledgments

This work was supported by NSF IIS-0916555 and a Google Faculty Research Award. We thank Gary Huang, Jerod Weinman, Ariel Prabawa, Flavio Fiszman, Jonathan Tiao, and Sammy Hajalie for their help and the reviewers for their constructive feedback.

References

[1] vis-www.cs.umass.edu/GLOC/.

[2] vis-www.cs.umass.edu/lfw/part_labels/.

[3] P. Arbeláez, B. Hariharan, C. Gu, S. Gupta, L. Bourdev, and J. Malik. Semantic segmentation using regions and parts. In *CVPR*, 2012.

[4] E. Borenstein, E. Sharon, and S. Ullman. Combining top-down and bottom-up segmentation. In *CVPR Workshop on Perceptual Organization in Computer Vision*, 2004.

[5] S. M. A. Eslami, N. Heess, and J. Winn. The shape Boltzmann machine: A strong model of object shape. In *CVPR*, 2012.

[6] S. M. A. Eslami and C. K. I. Williams. A generative model for parts-based object segmentation. In *NIPS*, 2012.

[7] X. He, R. Zemel, and M. Carreira-Perpinán. Multiscale conditional random fields for image labeling. In *CVPR*, 2004.

[8] X. He, R. Zemel, and D. Ray. Learning and incorporating top-down cues in image segmentation. In *ECCV*, 2006.

[9] G. E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002.

[10] G. B. Huang, V. Jain, and E. Learned-Miller. Unsupervised joint alignment of complex images. In *ICCV*, 2007.

[11] G. B. Huang, H. Lee, and E. Learned-Miller. Learning hierarchical representations for face verification with convolutional deep belief networks. In *CVPR*, 2012.

[12] G. B. Huang, M. Mattar, H. Lee, and E. Learned-Miller. Learning to align from scratch. In *NIPS*, 2012.

[13] G. B. Huang, M. Narayana, and E. Learned-Miller. Towards unconstrained face recognition. In *CVPR Workshop on Perceptual Organization in Computer Vision*, 2008.

[14] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[15] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, 2009.

[16] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001.

[17] Q. V. Le, M. A. Ranzato, R. Monga, M. Devin, K. Chen, G. S. Corrado, J. Dean, and A. Y. Ng. Building high-level features using large scale unsupervised learning. In *ICML*, 2012.

[18] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang. A tutorial on energy-based learning. In G. Bakir, T. Hofman, B. Schölkopf, A. Smola, and B. Taskar, editors, *Predicting Structured Data*. MIT Press, 2006.

[19] K. Lee, D. Anguelov, B. Sumengen, and S. Gokturk. Markov random field models for hair and face segmentation. In *FG*, 2008.

[20] J. Malik, S. Belongie, J. Shi, and T. Leung. Textons, contours and regions: Cue integration in image segmentation. In *ICCV*, 1999.

[21] D. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using brightness and texture. In *NIPS*, 2002.

[22] V. Mnih, H. Larochelle, and G. Hinton. Conditional restricted boltzmann machines for structured output prediction. In *UAI*, 2011.

[23] K. P. Murphy, Y. Weiss, and M. I. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *UAI*, 1999.

[24] M. Rastegari, A. Farhadi, and D. Forsyth. Attribute discovery via predictable discriminative binary codes. In *ECCV*, 2012.

[25] R. Salakhutdinov and G. Hinton. Deep Boltzmann machines. In *AISTATS*, 2009.

[26] L. Saul, T. Jaakkola, and M. Jordan. Mean field theory for sigmoid belief networks. *Journal of Artificial Intelligence Research*, 4:61–76, 1996.

[27] C. Scheffler, J. Odobez, and R. Marconi. Joint adaptive colour modelling and skin, hair and clothing segmentation using coherent probabilistic index maps. In *BMVC*, 2011.

[28] P. Smolensky. Information processing in dynamical systems: Foundations of harmony theory. *Parallel distributed processing: Explorations in the microstructure of cognition*, 1:194–281, 1986.

[29] N. Wang, H. Ai, and S. Lao. A compositional exemplar-based model for hair segmentation. In *ACCV*, 2011.

[30] N. Wang, H. Ai, and F. Tang. What are good parts for hair shape modeling? In *CVPR*, 2012.

[31] L. Wolf, T. Hassner, and Y. Taigman. Effective unconstrained face recognition by combining multiple descriptors and learned background statistics. *IEEE-TPAMI*, 33(10):1978–1990, 2011.

[32] Y. Yacoob and L. Davis. Detection and analysis of hair. *IEEE-PAMI*, 28(7):1164–1169, 2006.