Multi-Objective Convolutional Learning for Face Labeling

Sifei Liu, UC Merced
Jimei Yang, UC Merced
Chang Huang, Baidu Research
Ming-Hsuan Yang, UC Merced

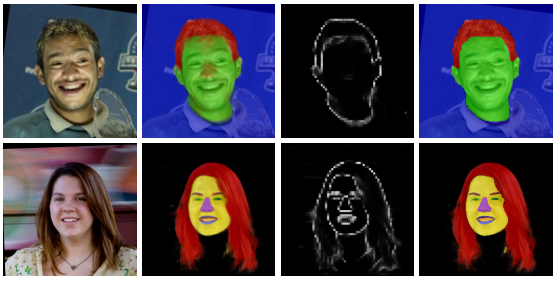
Abstract

This paper formulates face labeling as a conditional random field with unary and pairwise classifiers. We develop a novel multi-objective learning method that optimizes a single unified deep convolutional network with two distinct non-structured loss functions: one encoding the unary label likelihoods and the other encoding the pairwise label dependencies. Moreover, we regularize the network by using a nonparametric prior as new input channels in addition to the RGB image, and show that significant performance improvements can be achieved with a much smaller network size. Experiments on both the LFW and Helen datasets demonstrate state-of-the-art results of the proposed algorithm, and accurate labeling results on challenging images can be obtained by the proposed algorithm for real-world applications.

1. Introduction

Deep convolutional neural networks (CNNs) have been applied to image labeling and parsing problems [7, 4, 3, 13, 21]. As powerful end-to-end nonlinear classifiers, CNNs generate more discriminative representations compared to traditional methods based on hand-crafted features. Conditional random fields (CRFs) are another important class of image labeling models [14, 5, 6, 2] that carry out structured prediction by considering label dependencies and allow flexible use of pre-trained image features. We are concerned with combining CNNs and CRFs for image labeling by exploiting rich features from CNNs and structured output from CRFs [17, 23]. Considering a typical CRF energy function with unary and pairwise terms, a straightforward combination is to add CRF based structured losses on top of CNNs. Learning CNNs with structured loss, however, requires MAP inference of all the samples during training cycles. On one hand, it significantly increases computational cost while restricting the training flexibility. On the other hand, the direct combination of CNNs and CRFs with structured loss may not guarantee convergence.

This paper presents a novel learning method by decomposing the structured loss into two distinct, non-structured losses: softmax loss for the unary term and logistic loss for the pairwise term. The training process is carried out through a multi-objective optimization, which minimizes the losses of unary and pairwise terms respectively through a unified convolutional network. Weight sharing is enforced between them so that the network is strengthened by learning from both objectives. Compared to structured loss CNNs, our method has two advantages. First, the training process is as efficient as existing CNNs with non-structured losses. Second, by converting the pairwise term into a logistic loss (edge versus non-edge), semantic image boundaries are learned for effective labeling. Our model is trained on patches for flexibility. During the test stage, by making some simple adjustments, we can apply our patch model directly to a full image without patch cropping for efficient pixel-wise label prediction. We apply the proposed learning algorithm to a practical problem, face labeling that assigns every pixel a facial component label, e.g. skin, nose, eyes and mouth. Two examples are shown in Figure 1. Compared to facial landmarks, face labeling provides a better intermediate representation for many face analysis, synthesis and editing tasks.



(a) input (b) unary (c) edge (d) label

Figure 1. Face labeling on the LFW [10] and Helen [1] (a) input images (b) pixel-wise label likelihoods (c) semantic edge maps (d) face labeling results. Our algorithm first generates pixel-wise label likelihoods and semantic edge maps, which are combined in a CRF energy function to generate face labels. The images in (b) are soft labels (probabilistic outputs) and images in (d) are hard labels (excluding hair) which are shown in different colors. While the pixel-wise maps alone are effective for labeling, the use of edge maps further facilitates delineating the details, especially near the class boundaries.

Faces are highly structured visual patterns. For image labeling, we integrate a global facial prior into our learn-

ing model. The global facial prior is estimated by transferring labeling masks from exemplars through landmark detection [20]. Unlike existing methods [20, 26, 12] that use this nonparametric prior at the inference stage, our method uses it as additional input channels, other than raw RGB image intensities, to train a CNN. We show that this nonparametric prior significantly reduces the size of CNN in terms of both parameters and connections. In other words, it provides strong regularizations for CNN training to facilitate lightweight architectures.

The proposed face labeling algorithm is evaluated on two challenging benchmark datasets with 3 classes (LFW) [9] and 11 classes (Helen) [20]. Experimental results show that our algorithm performs favorably against state-of-the-art methods. We also present hair parsing results that can be generated simultaneously from the unified framework, which is more challenging and rarely addressed by existing methods.

The contributions of this paper are summarized as follows:

- a multi-objective convolutional learning method is developed for image labeling problems by decomposing the structured loss of CRFs into two distinct, non-structured losses, and optimizing a single unified CNN model with weight sharing;
- a nonparametric facial prior is introduced to CNN training that significantly reduces the network size;
- an efficient testing method is proposed to ensure fast, full-size labeling.

2. Related Work and Problem Context

Face labeling. To parse an input face image into semantic regions, e.g. eyes, nose and mouth for further processing, numerous methods have been developed that define a set of landmarks along face contours and facial components [25].

However, there are several issues with such facial landmark based representations. First, they are sensitive to pose, shape, illumination variations and occlusions. Second, there are usually no landmarks defined in the forehead and hair regions that are also important for applications such as face and hair editing [20].

Face labeling instead provides a more robust representation by assigning a semantic label to every pixel of a face image. Recently, several face labeling algorithms have been proposed based on CRFs [24, 10], deep learning [13] and exemplars [20]. Warrell and Prince [24] use a family of multinomial priors to model facial structures and a CRF for labeling facial components. In [10], Kae et al. model the face shape prior with a restricted Boltzmann machine and combine it with a CRF for labeling with 3 classes (background, face and hair). Unlike our approach, these two methods train classifiers based on hand-crafted image features as the unary terms of CRFs. Luo et al. [13] propose a deep learning based hierarchical face parsing method by combining several separately trained models, in which only facial components are labeled. In [20], Smith et al. develop a method to parse facial components and skin by transferring labeling masks from aligned exemplars. Compared to existing approaches, we propose an end-to-end unified model that generates complete labeling of facial components, skin and hair in a single pipeline.

Combining CNNs with graphical models. Our multi-object convolutional learning method is related to recent works [4, 15, 17, 23, 17, 23] that combine CNNs with graphical models for structured prediction problems. Farabet et al. [4] combine multiscale CNNs with a region tree structure for scene parsing. Specifically, they train CNNs in an unsupervised layer-wise manner from multiple scales. The learned multiscale image features are then used to train region-wise classifiers for label prediction in a pre-constructed segmentation tree. This is a typical two-step approach that sequentially trains a CNN and a graphical model.

Other than sequential combination, joint training of CNNs and graphical models have been reported in several papers [15, 17, 23]. Ranftl et al. [17] combines a variational energy model with CNNs for foreground/background image segmentation. The variational model used in [17] can be considered as a regularization of CRF labeling models. Three CNNs for unary, vertical pairwise and horizontal pairwise terms are trained separately without weight sharing. This method requires more memory during the testing stage and is more computationally expensive. In contrast, our method trains a single CNN with two distinct losses (one for unary and the other for pairwise) through multi-objective optimization. The learned model is thus lightweight and fast to evaluate. The joint training approach has also been applied to human pose estimation. In [23], a CNN model is used to train part detectors (unary) and part likelihood maps are then combined with image input to train pairwise spatial models between parts. Although using a single CNN, it still requires two-step training while our algorithm generates unary and pairwise terms simultaneously. We note that the above-mentioned approaches have not been applied to face labeling problems, especially combined with a nonparametric prior.

3. Multi-Objective Convolutional Learning

We formulate the problem of labeling a face image $\mathbf{X}$ as a CRF model $P(\mathbf{Y}|\mathbf{X}) = \frac{1}{Z}\exp(-E(\mathbf{Y},\mathbf{X}))$ where Z is the partition function and $\mathbf{Y}$ is a set of random variables $y_i \in \mathbf{Y}$ defined on every pixel i. Each variable y_i takes a value from a set of labels $\{\ell = 1, 2, .., K\}$. To consider the label dependencies, we introduce a 4-connected graph $(\mathcal{V}, \mathcal{E})$ where each node represents one pixel $i \in \mathcal{V}$ and edges represent the connections between any two ad-

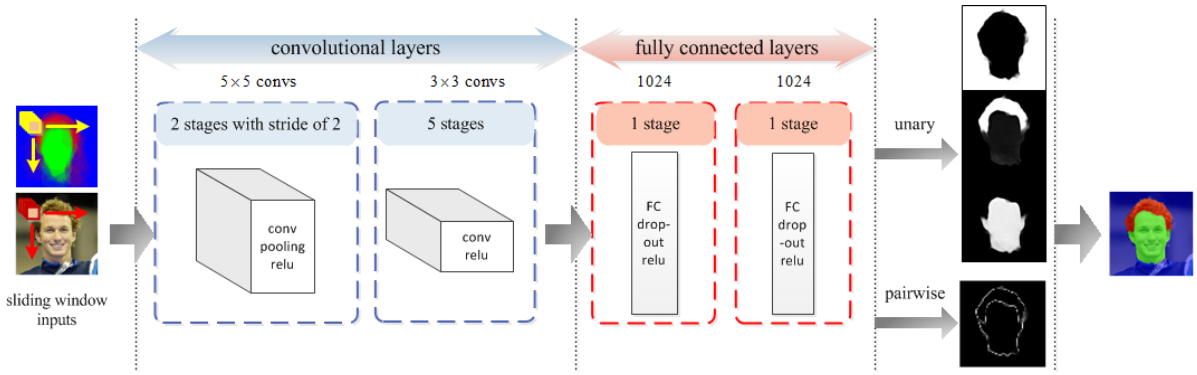


Figure 2. Proposed CNN classifier with sliding window based inputs.

jacent pixels $i, j \in \mathcal{E}$. Therefore, the CRF model can be expressed as a energy function $E(\mathbf{Y}, \mathbf{X})$ with two data-dependent terms:

$$E(\mathbf{Y}, \mathbf{X}) = \sum_{i \in \mathcal{V}} E_u(y_i, \mathbf{x}_i) + \lambda \sum_{(i,j) \in \mathcal{V}} E_b(y_i, y_j, \mathbf{x}_{ij}). \quad (1)$$

The unary term $E_u(y_i, \mathbf{x}_i)$ measures the assignment cost of variable y_i based on the image patch $\mathbf{x}_i$ centered at the pixel i and the pairwise term $V(y_i, y_j, \mathbf{x}_{ij})$ encodes the consistency cost of adjacent variables y_i, y_j given their overlapping patch $\mathbf{x}_{ij}$. In addition, λ is the mixing constant. We introduce a multi-class classifier $P_u(y_i = \ell | x_i, \omega_u)$ to express the label assignment cost for the unary term,

$$E_u(y_i, \mathbf{x}_i, \omega_u) = -\log P_u(y_i = \ell | \mathbf{x}_i, \omega_u). \quad (2)$$

To measure the consistency of two adjacent pixels i, j in the pairwise term, we introduce a new label $z_{ij} = 1$, if $y_i \neq y_j$ and $z_{ij} = 0$, otherwise. Thus the pairwise term is defined by the output of a binary classifier $P_b(z_{ij} = 1 | \mathbf{x}_{ij}, \omega_u)$,

$$E_b(y_i, y_j, \mathbf{x}_{ij}, \omega_b) = -\log P_b(z_{ij} = 1 | \mathbf{x}_{ij}, \omega_b). \quad (3)$$

In this work, we use CNNs with 9 layers for both unary and pairwise classifiers as they provide end-to-end predictions without using hand-crafted features.

Learning CNN parameters ω_u and ω_b jointly with the CRF model is difficult as the process needs to explore not only the combinatorial labeling space but also the large parameter space. To avoid this problem, an obvious approach is to train two independent CNNs for the unary and pairwise terms, respectively. We note that both CNNs are based on local image patches and should share very similar features in the lower layers [19]. In addition, the potentially large set of parameters from two CNNs may cause overfitting problems. In this paper, we propose to learn a single unified CNN for both unary and pairwise classifiers. By sharing all the features within a single CNN, the two classifiers are able to enjoy better generalization ability and higher computational efficiency.

We define two distinct loss functions for unary and pairwise classifiers, respectively. We denote the parameters of the shared CNN network by ω, and the feature response extracted from the topmost intermediate layer of CNN by $h_i = h(\mathbf{x}_i, \omega)$. Thus, the output of the unary classifier is given by a softmax function,

$$P_u(y_i = \ell | h_i, \omega_u) = \frac{\exp((\omega_u^\ell)^\top h_i)}{\sum_{\ell=1}^{K} \exp((\omega_u^\ell)^\top h_i)}, \quad (4)$$

where ω_u^ℓ represents the parameters for the ℓ-th class. Accordingly, the softmax loss for unary term is

$$L_u(y_i, \mathbf{x}_i, \omega, \omega_u) = -\log P_u(y_i = \ell | h_i, \omega_u). \quad (5)$$

On the other hand, the output of pairwise classifier is given by a logistic function,

$$P_b(z_{ij} = 1 | h_i, \omega_b) = \frac{1}{1 + \exp(-\omega_b^\top h_i))}, \quad (6)$$

and accordingly, the logistic loss for pairwise term is

$$L_b(z_{ij}, \mathbf{x}_{ij}, \omega, \omega_b) = -\log P_b(z_{ij} = 1 | h_i, \omega_b). \quad (7)$$

Based on these two loss functions (4) and (6), we train the unified CNN through multi-objective optimization,

$$\min_{\omega} \{O_u(\omega, \omega_u), O_b(\omega, \omega_b)\},$$
$$\begin{cases} O_u(\omega, \omega_u) = \mathbb{E}(\sum_{i \in \mathcal{V}} L_u(y_i, \mathbf{x}_i, \omega, \omega_u)) + \Psi(\omega, \omega_u) \\ O_b(\omega, \omega_b) = \mathbb{E}(\sum_{i,j \in \mathcal{E}} L_b(z_{ij}, \mathbf{x}_{ij}, \omega, \omega_b)) + \Phi(\omega, \omega_b) \end{cases} \quad (8)$$

where $O_u(\omega, \omega_u)$ is the expected loss $\mathbb{E}(\cdot)$ for the unary classifier and $O_b(\omega, \omega_b)$ is the expected loss for the binary classifier over all the training samples. In addition, $\Psi(\omega, \omega_u)$ and $\Phi(\omega, \omega_b)$ are regularization terms. The network is updated through combining gradients of both the softmax and logistic loss functions for backpropagation.

This multi-objective CNN has two main advantages: First, the convolutional network generates expressive representations at lower levels (layers that are close to the input

end) that can be utilized for both unary and pairwise model regressions. Second, the unified network can be learned by backpropagating errors from both outputs jointly such that the network can learn features that are highly adaptive to both objectives. The shared model also alleviates overfitting problems and reduces the overall model size such that both training and testing can be carried out efficiently.

3.1. CNN Architecture

Since CNNs usually operate on a patch level centered at each pixel, the labeling pipeline is based on a sliding window input [18, 23] with overlapping patches, as shown in Figure 2. We propose an architecture similar but deeper [19] than that of [11], with 7 convolutional and 2 fully connected layers. The inputs are 72×72 single scale patches which are passed to two top consecutive convolutional units with a filter of 5×5, where each convolutional layer is followed by one max pooling layer with a downsampling stride of 2. Following that is another stack of small convolutional units with a receptive field of 3×3 and no pooling layer. All the layers are equipped with a rectification (ReLU) nonlinearity and a local response normalization (LRN) layer.

3.2. Nonparametric Prior

We introduce a nonparametric prior as global regularization for face labeling, which is estimated by transferring label masks from exemplars. Given a test image, five facial keypoints on eyes, nose tip and two mouth corners are detected using the method in [22]. We construct a shape subspace based on principal component analysis from ground truth facial keypoints in the exemplar set. Thus, the facial keypoints of the test image are used to find a set of exemplar images with most similar shapes on that subspace. For any pair of test-exemplar images, corresponding keypoints are used to estimate similarity transformation parameters so that the ground truth label masks of exemplars can be aligned with the test image for nonparametric shape prior. Our nonparametric prior is simply the average of all the aligned ground truth label masks. The generated nonparametric prior is then used as an additional input channel for training the CNN. A typical labeling improvement using this prior is shown on the right side of Figure 3. The CNN trained on image patches incorrectly labels the face on the upper right image according to its local content while the CNN trained on both prior and image patches is able to reject the false label assignments. Moreover, the prior input introduces regularization to the CNN so that the training could converge faster. We will also show that using this prior leads to significant reduction of the network size without degrading performance.

3.3. Adaptive Inference

Submodular energy function. In the testing stage, the labeling process involves evaluating the learned CNN for both unary and pairwise terms and CRF inference. Figure 2 demonstrate the inference pipeline. Given pixel-wise label likelihood maps for the unary term $E_u(y_i)$ and edge map for the pairwise term $E_b(y_i, y_j)$, we convert the original energy function to a submodular one so that the GraphCut algorithm can be used for efficient inference,

$$\min E(\mathbf{Y}) = \sum_{i \in \mathcal{V}} E_u(y_i) + \sum_{i,j \in \mathcal{V}} E_b(y_i, y_j) \mathbb{I}(y_i \neq y_j), \tag{9}$$

where $\mathbb{I}(\cdot)$ is the indicator function.

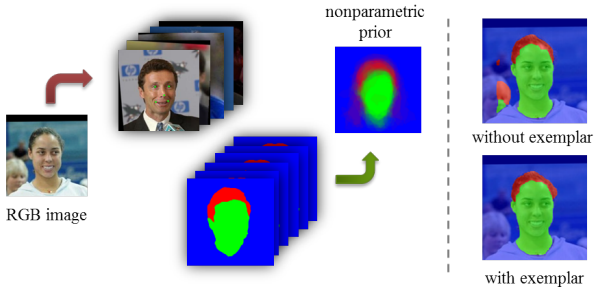


Figure 3. An nonparametric prior is proposed based on label transfer, as shown on the left. A typical labeling improvement is shown on the right. The CNN trained on image patches without exemplars incorrectly labels the face on the upper left part according to its local content while the CNN trained on both prior and image patches is able to reject the false label assignments.

Adapting the patch CNN to the full image. To generate pixel-wise label likelihood maps efficiently, we make some adjustments for the CNN trained on patches. Our CNN architecture consists of more layers but has a smaller filter size, compared to that in [11]. Thus, we have a considerate size of overall receptive field and more nonlinearity of the decision function, without increasing the number of parameters. Specifically, we use fewer pooling layers (only in the first two convolutional units) to preserve more spatial information from the input image. In the training stage, we sample a group of patches centered with randomly selected pixels for each training image. The network is supervised through the corresponding labels y and z respectively for unary and pairwise training, and updated by mini-batch gradient descent.

Since convolutional operations share computations between overlapped patches, the sliding-window pipeline for each pixel of a test image is computationally redundant. We propose to use an efficient *patch-training and image-testing* strategy introduced in [18] by replacing the fully connected (FC) layers with equivalent convolutional layers, and setting the filter size as 1×1 (See Figure 2). We then apply the fully-convolutional model directly to a test image. Note that a test image is proper padded to ensure that every pixel corresponded "window" can be covered in order to generate the exactly equivalent result. Both labeling and edge probability maps can be generated by one full image forward

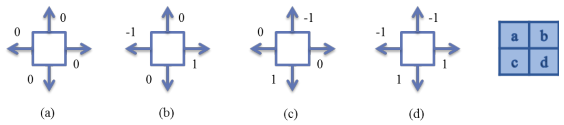


Figure 4. Generating a twice upsampled output map. The original image in (a) is shifted with additional 3 versions (b-d) along the x- and y-axis, with a step of 1. The high-resolution labeling is obtained by interlacing them in the way shown on the right, with a 2×2 block. The upsampled map layouts are shown on the right.

propagation, which is much faster than applying the model thousand times to sliding windows.

One problem with the proposed full image testing approach is that the size of the output maps is smaller than that of the original image due to the downsampling strides in the max pooling layers. Most existing approaches upsample the low-resolution map to the image size [4]. We propose to obtain the original-sized output maps by forward propagating a group of input images, generated by shifting the original input image with one or more pixels (depending on the zooming factor) on the x and y axis, as described in [16]. The way of generating an upsampled output map with a zooming factor of 2 is illustrated in Figure 4. In this work (two pooling layers with downsampling stride of 2), 4×4 times of forward propagations from shifted input images generate an upsampled output map with a zooming factor of 4 in a similar way. Since the size of the maps may still be inconsistent with that of the original image due to the border effect of convolutional operation, the final output map can be obtained by rescaling them to the exact input image size. Note that 16 passes of forward propagation is still much faster than applying the convolutional network to patches at each pixel of an image.

4. Experimental Results

We evaluate the proposed algorithm on two different benchmark datasets with different face labeling tasks. We show that it applies to both tasks and performs favourably against state-of-the-art methods with the same framework and experimental settings. Specifically, we demonstrate that both the multi-objective approach and the nonparametric prior improve the performance in all aspects compared to a per-pixel CNN classifier. We validate that the nonparametric prior introduces regularization to the network by reducing the number of network parameters and connections.

4.1. Datasets and Settings

Datasets. We use the LFW [10] dataset which has been used by recent methods for face labeling [24, 13]. However, the image subsets that were used for training and testing by these two methods are not available to the public. Kae *et al.* [10] report their 3-classes face labeling results on a released subset of LFW. For fair comparisons, we choose to conduct our first labeling experiment on the same subset of images, named *LFW part labels database (LFW-PL)*, with the same evaluation criteria.

The LFW-PL set contains 2927 face images of 250×250 pixels acquired in unconstrained environments. All of them are manually annotated with skin, hair and background labels using superpixels. This dataset is divided into a training set with 1500 images, a testing set with 927 images, and a validation set with 500 images. The validation set is used to generate the nonparametric prior for each training and testing image, as described in Section 3.

We use the HELEN [1, 20] dataset containing face labels with 11 classes for the second set of experiments. It is composed of 2330 face images of 400×400 pixels with labeled facial components generated through manually-annotated contours along eyes, eyebrows, nose, lips and jawline. The hair region, not considered in the labeling categories in [20], is annotated through a matting algorithm. The dataset is also divided into a training set (corresponding to the exemplar set in [20]) with 2000 images, a testing set with 100 images, and a validation set (corresponding to the tuning set in [20]) with 300 images.

Network Configurations. Similar network configurations are applied to the *LFW-PL* and *Helen* datasets for face labeling. As mentioned in the Section 3.1, we use a single-scale patch input with a size of 72×72 pixels in order to keep a proper receptive field. Compared to a multi-scale input [4, 23], the single-scale configuration is easy to adapt from patch-based training to image-based testing. The released images in the *LFW-PL* dataset are coarsely aligned using the congealing alignment method [10]. We align the images of the *HELEN* dataset to a canonical position by detecting five facial keypoints using [22], and computing the similarity transformation using least squares minimization. To adapt to the receptive field to the input patch size, we further resize the images and evaluate them with a size of 250×250 pixels.

The CNN training procedure is carried out using mini-batch gradient descent with the momentum, weight decay, dropout ratio, and batch size are set to 0.9, 5×10^{-4}, 0.5, and 50, respectively. All are kept unchanged throughout the training procedures. The learning rate is initially set to 10^{-3} and is manually decreased by a factor of 10 when the loss on the validation set starts fluctuating.

We use CNNs without pairwise and nonparametric prior as a baseline evaluation, denoted as **S-CNNs**. We evaluate our multi-objective approach with respect to: (a) unary term (**MO-unary**) with a softmax probabilistic output; (b) inference results from both unary and pairwise terms through GraphCut, denoted as **MO-GC**. We denote a suffix of **"with prior"** for the approaches with the nonparametric prior.

Sampling. In the training stage, patch sampling is generally based on a random criterion. However, the number of

(a) image (b) no prior (c) prior (d) no prior (e) prior

Figure 5. Comparison for usage of nonparametric prior. (a) test images; (b) labeling results generated by MO-GC; (c) labeling results generated by MO-GC with nonparametric prior. (d) semantic edge generated by MO-GC; (e) semantic edge generated by MO-GC with nonparametric prior. Best viewed in color.

Table 1. Overall per-pixel accuracy on the *LFW-PL* dataset with channel numbers of two FC layers setting as 4096 and 1024. The F-measure of skin (F-skin), hair (F-hair) and background (F-bg) are also presented.

(%)	accuracy	F-skin	F-hair	F-bg
S-CNNs	92.92	90.07	73.73	95.18
MO-unary	93.45	91.45	78.03	95.84
MO-GC	93.77	91.95	79.06	96.03
S-CNNs with prior	94.25	92.79	77.18	96.63
MO-unary with prior	94.94	93.64	79.95	97.02
MO-GC with prior	**95.12**	**93.93**	**80.70**	**97.10**

Table 2. Overall accuracy on LFW-PL with comparison to [10]. Note that the evaluation of GLOC is based on a superpixel-wise accuracy, and ours are based on a per-pixel evaluation.

(%)	GLOC (SP)	MO-unary	MO-GC
accuracy	94.95	95.03	**95.24**
error reduction	25.41	26.59	**29.69**

patches from rare classes may be insufficient for training an effective network. This is particularly obvious with semantic edges and some facial components such as eyes and lips, which take a relatively small portion of pixels. Therefore, we apply a two-stage training procedure with different sampling approaches. We first train the convolutional network by keeping a certain ratio of the number of patches between one or more rare classes and the others, so that a sufficient number of samples can be drawn for the rare classes. We then apply the globally random sampling on the second stage for fine-tuning to ensure the network adapts to the natural distributions of classes.

4.2. LFW-PL

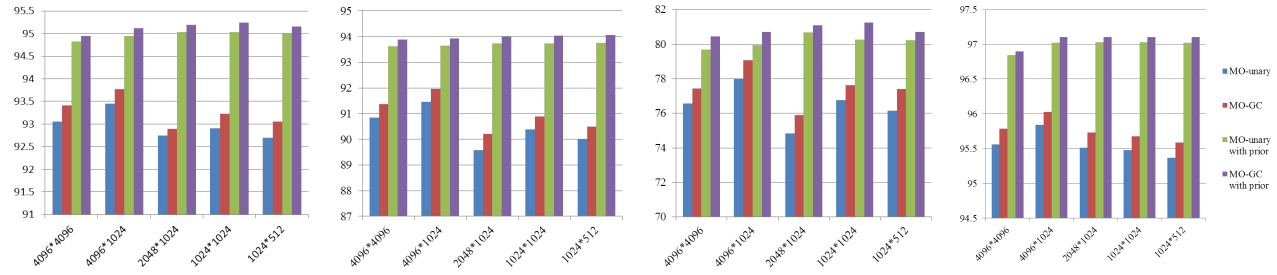
We first show results on face labeling of skin, hair and background. In this task, the classes are relatively balanced in the number of pixels, and we randomly sample 12 batches from each training image. We additionally sample 12 batches with the ratio of non-edge and edge setting to 1.2 (this step is removed in the fine-tuning stage). We also randomly generate affine transformations as augmentation over patches [4] to one random batch. This is easy to apply to a patch-based training approach, and is particularly effective for increasing the variation of training samples, especially when the number of training images is small.

In table 1, we test a series of approaches with the channel numbers of the two FC layers as 4096 and 1024, and evaluate the results with respect to per-pixel accuracy and F-measure of each class. The first 3 rows show the approaches without nonparametric prior input, while the lower 3 rows are those using it. Overall, the nonparametric prior significantly improves the results when compared with all corresponding approaches. We specifically compare the labeling and semantic edge results in terms of the nonparametric prior in Figue 5. By comparison between networks with (shown in (c) and (e)) and without (shown in (c) and (e)) the nonparametric prior, we observe that the labeling is improved in case of blurry hair region (Figure 5(a)), blurry face (Figure 5(b)), multiple person (Figure 5(c)) and moustache (Figure 5(c)), through introducing the prior. Moreover, the proposed multi-objective approach (beginning with MO) generally outperform the CNN classifiers (S-CNNs and S-CNNs with prior). The inference step can further improve the performance for all tested approaches.

Another major improvement of the multi-objective approach can be observed from the comparison between S-CNNs and MO-unary, as shown in the row 1 vs.2 and 4 vs.5 in Table 1. While both of them are evaluated from the labeling probabilities, the MO-unary network contains an additional output that learns the semantic edges. Namely, even if two networks are trained under the same conditions (network configurations, nonparametric prior, etc.), the one with a supervised semantic edge learning generates results in more expressive representations by back-propagating information of class boundaries.

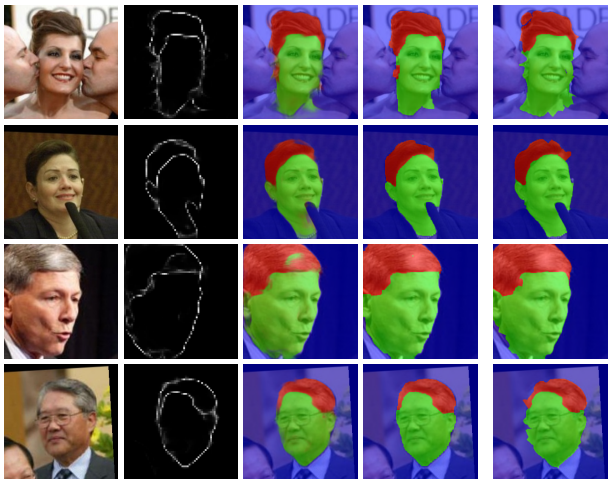
Network regularization. By introducing a nonparametric prior as an additional input, the network size can be significantly reduced without degrading the performance. We use different settings of the FC layers as they usually take a large portion of weights and connections in a deep CNN architecture. The combination of channel numbers regarding to two FC layers is denoted by $a * b$, where a and b are neural numbers for the first and the second FC layer, respectively. Four network configurations, as listed in Table 3, are applied for evaluation of different model sizes, as shown in Figure 6.

With nonparametric prior (colored with green and purple), the performance of the smaller networks (e.g. $1024 * 1024$ and $1024 * 512$) is comparatively similar to that of the



(a) accuracy (b) F-skin (c) F-hair (d) F-bg

Figure 6. We show the network regularization by introducing the proposed nonparametric prior as an additional input. Four FC settings associated with Table 3 are used to control the size of the network, as shown in the X-axes.



(a) image (b) pairwise (c) unary (d) GC (e) ground truth

Figure 7. Face labeling results and semantic edge maps from LFW-PL dataset. (a) test images; (b) pairwise term output; (c) unary term output; (d) labeling result by GraphCut inference, denoted as GC; (e) ground truth. Best viewed in color.

Table 3. Four settings of channel numbers for the two FC layers, and their corresponding model size in MB.

FC	4096*	4096*	2048*	1024*	1024*
$a * b$	4096	1024	1024	1024	512
size (MB)	163	119	65	38	36

large-size networks (e.g. $4096 * 4096$ and $4096 * 1024$). With the inference step (MO-GC with prior), the network of $1024 * 1024$ achieves the highest overall accuracy, while the network of $1024 * 512$ achieves the highest F-score for the class of skin. On the contrary (colored with blue and dark red), the networks without nonparametric prior generally have a worse performance when decreasing the model size. For the networks less than 119 MB ($4096 * 1024$), the per-pixel accuracies are no higher than 93%.

Comparison to GLOC. We compare the results with GLOC [10] by following their evaluation of overall accuracy and error reduction with respect to a standard CRF with features in Huang *et al.* [8], as shown in Table 2. We apply the setting of FCs with $1024 * 1024$ which achieves the best performance. Note that a major difference compare to [10] is that it applies a superpixel-wise accuracy since it is a superpixel based method, while we use a per-pixel accuracy evaluation since our approach outputs a per-pixel labeling map. For GLOC, the per-pixel evaluations may be slightly different, which however is not reported in the paper.

We illustrate for several images the unary and pairwise output maps, as well as the inference results in Figue 7. Beside an accurate probability output, as shown in Figure 7(c), our approach also generates clear and accurate semantic edges in Figure 7(b), which are useful for many vision applications but have not been addressed in prior face labeling approaches. Through inference addressed in Section 3.3, our final labeling results are quite promising and are able to handle many challenge cases, such as multiple persons (first row) and occlusion (second row).

Figure 7(e) show the ground truth labeling for selected examples, we notice that the superpixel labeling proposed by [10] does not generate accurate annotations. Obvious disconnections along class boundaries are shown in the last two rows. On one hand, it introduces noises to the supervised CNN training, on the other hand, the superpixel-wise evaluation in [10] does not reveal the real accuracy. For instance, our results in Figure 7(d) contains a certain number of incorrect label assignments evaluated on the ground truth in Figure 7(e). However, they are visually even better than the ground truth, particularly along the class boundaries.

4.3. HELEN

We also show results on labeling of HELEN with two eyes, two eyebrows, nose, upper and lower lips, inner mouth, facial skin and hair. Unlike LFW-PL, some facial components are rare classes (e.g. eyes, lips, etc.) and therefore the two-stage sampling strategy proposed in Section 3.3 is applied. Instead of sampling the first 12 batches in a random way as in LFW-PL, we propose to firstly separate the labels as foreground (containing all facial components, skin and hair) and background. We then sample the first 11 batches randomly from foreground and the remaining one from background. In this way, the foreground is sufficiently trained in the first stage, and a natural foreground

Figure 8. Face labeling on the *HELEN* [1]. We visualize the proposed labeling results, shown in the second row, for the 11 classes network containing hair labeling. The ground truth is shown on the third row. For facial components and skin, we show a labeling map generated by the inference step. For hair, we visualize the probability output (with values ranging from 0 to 1) from the unary term to show a visually natural results. Best viewed in color.

Table 4. Evaluations on HELEN. We use float numbers instead of percentages to keep consistent on the numerical percision with [20]. For comparison, eyes, brow and mouth all are computed by combining related categories, and the "overall" denotes all facial components excluding facial skin.

methods	eyes	brows	nose	in mouth	upper lip	lower lip	mouth all	facial skin	overall
Smith *et.al*	**0.785**	0.722	**0.922**	0.713	**0.651**	**0.700**	**0.857**	0.882	0.804
Ours, 11 classes	0.768	0.713	0.909	0.808	0.623	0.694	0.841	0.910	0.847
Ours, 10 classes	0.768	**0.734**	0.912	**0.824**	0.601	0.684	0.849	**0.912**	**0.854**

label distribution can be preserved. We repeat the same edge sampling and jitter generation strategy with LFW-PL. Specifically we train two models for HELEN: For the first model, we train a 11-class unified convolutional network, with the multi-objective approach with nonparametric prior as additional input. Therefore, we show that the hair labeling can be generated along with other facial labels, which is not addressed in prior work. For the second model, we merge the ground truth hair label with the background to train a 10-classes network using the same approach. In this way, a fair comparison with the work of [20] can be obtained. The FC layers are fixed to $4096 * 1024$.

Based on the same subset of images with same evaluation criteria, we simply report the results of [20]. In Table 4, a large variation in F-measure with respect to each facial component can be seen between [20] and the proposed approaches. While [20] bases the work on exemplar transfer, and obtains better results on relatively rare facial classes, such as eyes, nose and mouth, we outperform it in facial skin and the overall components. Specifically, we achieve an overall F-measure of 0.854, which is a noticeable improvement over the work of [20].

Table 4 shows that the labeling of hair regions, which is challenging and seldom addressed in existing facial component labeling methods, can be successfully generated together with other facial components by the proposed algorithm in a unified model. With hair labeling, this proposed method still performs well in overall facial components against the state-of-the-art method on the HELEN dataset. We notice in Figure 8 that the proposed approach generates accurate labeling results on each facial component (second row) compared to the ground truth (third row). Specifically, it generates visually pleasant labeling results in some challenging cases (even for human beings), as shown in the sixth and seventh column.

5. Conclusion

We propose a deep convolutional network that jointly models pixel-wise likelihoods and label dependencies through a multi-objective learning method. We introduce a nonparametric prior, combining it with the RGB image together as input to the network, and show that it provides a strong regularization to the network, so that a much smaller model can still achieve a competitive performance. Experiments on face labeling tasks validate that the proposed approach significantly improves the labeling performance and generates visually pleasant labeling results.

6. Acknowledgements

This work is supported in part by the NSF CAREER Grant #1149783, NSF IIS Grant #1152576, and the National Basic Research Program of China (973 program No. 2014CB340505).

References

[1] http://www.ifp.illinois.edu/ vuongle2/helen/. 1, 5, 8

[2] L. Bertelli, T. Yu, D. Vu, and B. Gokturk. Kernelized structural svm learning for supervised object segmentation. In *CVPR*, 2011. 1

[3] D. Ciresan, A. Giusti, L. Gambardella, and J. Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In *NIPS*, 2012. 1

[4] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *IEEE PAMI*, 35(8):1915–1929, 2013. 1, 2, 5, 6

[5] B. Fulkerson, A. Vedaldi, and S. Soatto. Class segmentation and object localization with superpixel neighborhoods. In *ICCV*, 2009. 1

[6] J. M. Gonfaus, X. Boix, J. V. D. Weijer, A. D. Bagdanov, J. Serrat, and J. Gonzalez. Harmony potentials for joint classification and segmentation. In *CVPR*, 2010. 1

[7] D. Grangier, L. Bottou, and R. Collobert. Deep convolutional networks for scene parsing. In *ICML 2009 Deep Learning Workshop*, 2009. 1

[8] G. Huang, M. Narayana, and E. Learned-Miller. Towards unconstrained face recognition. In *CVPR Workshop*, 2008. 7

[9] 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. 2

[10] A. Kae, K. Sohn, H. Lee, and E. Learned-Miller. Augmenting CRFs with Boltzmann machine shape priors for image labeling. In *CVPR*, 2013. 1, 2, 5, 6, 7

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 4

[12] C. Liu, J. Yuen, and A. Torralba. Nonparametric scene parsing via label transfer. *IEEE PAMI*, 33(12):2368–2382, 2011. 2

[13] P. Luo, X. Wang, and X. Tang. Hierarchical face parsing via deep learning. In *CVPR*, 2012. 1, 2, 5

[14] S. Martin, K. Pushmeet, and H. Derek. Learning CRFs using graph cuts. In *ECCV*, 2008. 1

[15] F. Ning, D. Delhomme, Y. LeCun, F. Piano, L. Bottou, and P. E. Barbano. Toward automatic phenotyping of developing embryos from videos. *TIP*, 14(9):1360–1371, 2005. 2

[16] P. Pinheiro and R. Collobert. Recurrent convolutional neural networks for scene parsing. *arXiv preprint arXiv:1306.2795*, 2013. 5

[17] R. Ranftl and T. Pock. A deep variational model for image segmentation. In X. Jiang, J. Hornegger, and R. Koch, editors, *Pattern Recognition*, Lecture Notes in Computer Science, pages 107–118. Springer International Publishing, 2014. 1, 2

[18] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013. 4

[19] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3, 4

[20] B. Smith, L. Zhang, J. Brandt, Z. Lin, and J. Yang. Exemplar-based face parsing. In *CVPR*, 2013. 2, 5, 8

[21] R. Socher, C. Lin, C. Manning, and A. Y. Ng. Parsing natural scenes and natural language with recursive neural networks. In *ICML*, 2011. 1

[22] Y. Sun, X. Wang, and X. Tang. Deep convolutional network cascade for facial point detection. In *CVPR*, 2013. 4, 5

[23] J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. *arXiv preprint arXiv:1406.2984*, 2014. 1, 2, 4, 5

[24] J. Warrell and S. J. Prince. Labelfaces: Parsing facial features by multiclass labeling with an epitome prior. In *ICIP*, 2009. 2, 5

[25] Z. Xiangxin and R. Deva. Face detection, pose estimation, and landmark localization in the wild. In *CVPR*, 2012. 2

[26] J. Yang, Y.-H. Tsai, and M.-H. Yang. Exemplar cut. In *CVPR*, 2013. 2