FACE ATTRIBUTE CLASSIFICATION USING ATTRIBUTE-AWARE CORRELATION MAP AND GATED CONVOLUTIONAL NEURAL NETWORKS

Sunghun Kang, Donghoon Lee, and Chang D. Yoo

Korea Advanced institute of Science and Technology
Department of Electrical Engineering
291 Daehak-ro, Yuseong-gu, Daejeon, Korea

ABSTRACT

This paper proposes a face attribute classification method based on attribute-aware correlation map and gated convolutional neural networks (CNN). The attribute-aware correlation map provides correlation information between pixel-location and attribute label, and each correlation map of an attribute provides information regarding regions where the relevant features should be extracted. Using the correlation maps of all the attributes, a number of most relevant face part regions are discovered. Based on the face part regions, gated columns of CNNs are simultaneously pre-trained on for face representations then fine-tuned for attribute classification. Here, each CNN column takes input from one of the regions discovered. The column of the CNN is gated such that in the backpropagation of the learning process, classification error due to less relevant attributes do not over influence the learning process. In the experiment, we manually labeled each image in the Labeled Faces in the Wild (LFW) benchmark dataset with 40 face attributes and obtained significant performance improvement over other state-of-the art methods.

Index Terms— Face attribute classification, Attribute classification, Convolutional Neural Network, Gated CNNs, Face representation

1. INTRODUCTION

Predicting the presence of various face attributes that includes biometric features as well as the expression and accessories worn can be conducive for various tasks involving tagging [1, 2], searching [1], ranking [3, 2], and face verification [4, 5]. and verifying a face. This prediction is often referred to as attribute classification, and variations in pose, illumination, and occlusion render it a difficult problem.

To consider many attributes simultaneously in an efficient and effective manner, a common framework is often conceived for multiple attribute classification. The framework should extract effective features from pertinent facial regions for each attribute considered, and classification should be conducted based on relevant features extracted [6, 7, 8, 9, 10]. It should be noted that the features and pertinent regions would be different depending on the attribute. For example to predict the presence of eyeglasses, relevant features near the eyes should be examined whereas to determine baldness, relevant features extracted differently from the features extracted from the eyes are extracted from the head and the forehead.

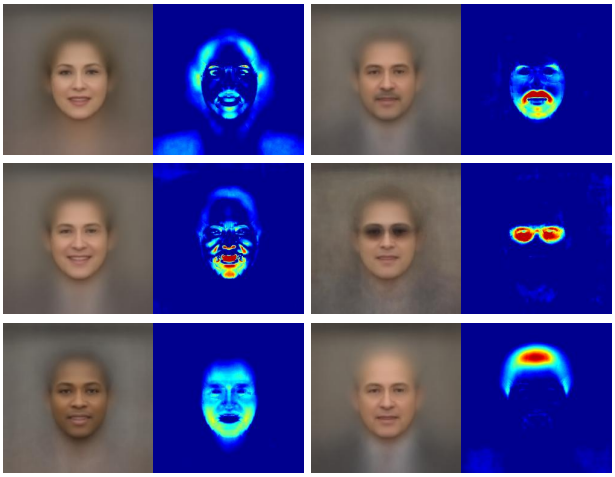


Fig. 1. The first and the third columns represent mean faces for different attributes while the second and fourth columns represent correlation maps between attributes and pixel values at different image location. Mean faces and attribute-aware correlation maps for six attributes are illustrated. The attributes in the first column (*female, smiling*, and *African American*) are correlated with global region, while the attributes in the second column (*mustache, sunglasses*, and *bald*) are correlated with specific local regions.

Fig. 1 shows examples of mean face and associated correlation map for six attributes(female, smiling, african american, mustaches, sunglasses, bald). As shown, different facial location provide different amount of information about an attribute.

For face attribute classification, this paper proposes a deep architecture composed of a number of different convolutional neural network (CNN) columns where each column extracts

relevant features from different facial regions. The outputs of the columns are gated and combined, such that the error is backpropagated to each column with different strength in the learning process. For a given attribute, different columns contribute differently in the prediction.

The different regions from which the CNN are to extract relevant features are pre-determined by discovering regions most correlated with the attributes. Using training data with associated attribute labels, the empirical cross-correlation between each pixel location and attribute can be determined. For each attribute, a correlation map can be constructed, and from the correlation maps of all attributes, a fixed number of regions most correlated with the attributes can be determined.

There have been a number of deep architecture based algorithms for attribute classification [11, 12, 13], and the proposed deep architecture distinguishes itself from others in two aspects: 1) the pre-determined regions are discovered in a systematic way such that it is highly correlated with the attributes and 2) the outputs of the columns of CNN are combined with gates, that the error is backpropagated to each column with different strength in the backpropagation of the learning process.

The remainder of the paper is organized as follows. Section 2 describes the details of the proposed attribute-aware correlation map for face part selection and gated CNNs for face representation. The experimental results are reported in Section 3, and the conclusions are presented in Section 4.

2. METHOD

The proposed method consists of attribute-aware correlation map for discovering facial regions correlated with face attributes, and gated CNNs for preventing over influencing classification error due to less relevant attributes.

2.1. Attribute-aware Correlation Map

In Algorithm 1, the procedure for obtaining attribute-aware correlation maps is described. Given a set of N face images $\{\mathbf{x}_i\}_{i=1}^N$, and their corresponding 68 facial landmark points $\{\mathbf{s}_i\}_{i=1}^N$ estimated using [14]. The face images is aligned using facial landmark estimates and is resized to a fixed size $W \times H$. The mean landmark is defined as the average of all the landmarks $\bar{\mathbf{s}} = \frac{1}{N}\sum_{i=1}^N \mathbf{s}_i$. To obtain correlation maps $\{I_{\text{corr}}^t\}_{t=1}^T$ for T attribute classes, we compute pixel-wise correlations over all image locations as described in Algorithm 1. To obtain a more precise results, pixel locations are indexed by a local index $(\delta w, \delta h, l_{\text{ref}})$ rather than a global index (w, h). Here, l_{ref} is the index of the closest mean landmark to (w, h) that is to be referred, and $(\delta w, \delta h)$ is the local coordinate of (w, h) to $\bar{s}^{l_{\text{ref}}}$. For all face images and corresponding landmark estimates, pixel values at the local index $(\delta w, \delta h, l_{\text{ref}})$ are extracted and concatenated to form $\mathbf{v}^{w,h} = (v_1^{x,y}, \cdots, v_N^{x,y})^\top$. The value at (w, h) for mean faces $\{I_{\text{mean}}^t\}_{t=1}^T$ and correlation maps $\{I_{\text{corr}}^t\}_{t=1}^T$ are determined by an average of $v_{i:y_i^t=1}^{w,h}$ and a correlation between $\mathbf{v}$ and $\mathbf{y}^t$, respectively.

Algorithm 1 Pixel-wise attribute-aware correlation maps.

Input: landmark estimates $\{\mathbf{s}_i\}_{i=1}^N$, face images $\{\mathbf{x}_i\}_{i=1}^N$, attribute labels $\{y_i^t\}_{i=1,t=1}^{N,T}$, mean landmark $\bar{\mathbf{s}}$.
Output: correlation map $\{I_{\text{corr}}^t\}_{t=1}^T$.
Procedure:

1: **for** $(w, h) = (1, 1), \cdots, (W, H)$ **do**
2: Compute local coordinate $(\delta w, \delta h, l_{\text{ref}})$ of (w, h):
$l_{\text{ref}} = \text{argmin}_l \, ||\bar{s}^l - (w, h)^\top||_2^2$,
$(\delta w, \delta h) = (w, h)^\top - \bar{s}^{l_{\text{ref}}}$.
3: Obtain pixel values from images:
$v_i^{w,h} = I_i(s_i^{l_{\text{ref}}} + (\delta w, \delta h)^\top), \quad i = 1, \cdots, N.$
4: Compute outputs at (w, h):
$I_{\text{corr}}^t(w, h) = \text{Corr}(\mathbf{v}^{w,h}, \mathbf{y}^t), \quad t = 1, \cdots, T.$
5: **end for**

Because $\{I_{\text{corr}}^t\}_{t=1}^T$ indicate the pixel-wise correlation to the each attributes, the following score for a function face part region R_p can be explicitly defined:

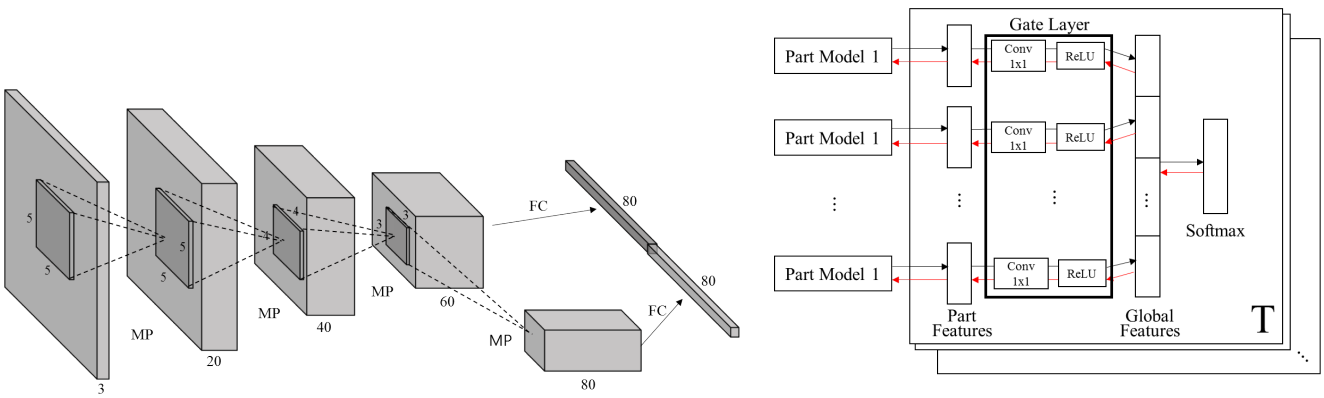
$$F(R_p) = \frac{1}{\text{Area}(R_p)} \sum_{(w,h):(w,h)\in R_p} \sum_{t=1}^{T} (I_{\text{corr}}^t(w, h)). \quad (1)$$

In Eqn. 1, $F(R_p)$ is defined as an average of correlation maps over face part region R_p. To localize P face part regions $\{R_p\}_{p=1}^P$ that maximizes $\sum_{p=1}^P F(R_p)$, we initialize 15 face part regions and iteratively refine their locations by weighted arithmetic mean.

2.2. Gated CNNs

In the learning procedure of the CNN columns loss signals over all attribute labels are summed and are propagated to update the parameters. Here, each attribute labels equally contributes for learning, and it might degenerates the performance. Because, for the given face part region, *e.g.* eye region, and the corresponding CNN column, the irrelevant attribute labels, *e.g.* mustache, is hard to be learned, and the loss signal from the relevant attribute labels will become dominant than the loss signal from the correlated attribute label, *e.g.* sunglasses. Loss signals from the irrelevant attribute labels disturb the model to be learned from the relevant attribute labels, thus, this should be considered in designing of the model.

Model architecture. The proposed gated CNN is depicted in Fig. 2–(b). The gated CNN consists of P CNN columns and is designed to neglect the loss signal from irrelevant attribute labels by using T gate layers assigned to each T attribute classes. Each gate layer consists of P gates which are



(a) Single CNN column architecture

(b) Gated CNN architecture

Fig. 2. The architectures of (a) single CNN column and (b) gated CNN, black and red arrows indicate forward pass and backward pass, respectively.

assigned for each face part. The gate consists of the 1×1 convolution layer and subsequent Rectified Linear Unit (ReLU), and its weight is automatically optimized in learning procedure to scale down and up the loss signals from irrelevant attribute labels and relevant attribute labels, respectively.

Each CNN column has four convolution layers with kernel sizes 5×5, 5×5, 4×4, 3×3, and four 2×2 max pooling layers as depicted in Fig. 2–(a). To extract both mid-level and high-level representation, fully-connected layer is attached on the 3rd, 4th convolution layer. Each CNN column utilizes the face part region which is output of attribute-aware correlation map.

Learning. Gated CNN is learned using error-back propagation. The parameters of gate layers are automatically optimized to provide appropriate strength of the loss signals from attribute labels. Given the input image $\{\mathbf{x_i}\}_{i=1}^{N}$ and corresponding attribute labels $\{\mathbf{y}_i\}_{i=1,t=1}^{N,T}$ the overall loss is defined as

$$L = \sum_{i=1}^{N}\sum_{t=1}^{T} y_i^t \log p(y_i^t|\mathbf{x}_i) + (1-y_i^t)\log(1-p(y_i^t|\mathbf{x}_i)), \quad (2)$$

where $p(y_i^t|\mathbf{x}_i) = \frac{1}{1+\exp(-\text{net}(\mathbf{x}_i))}$, and net$(\cdot)$ computes network outputs.

Classification. For classifying face attributes, the gate layers and softmax layers are removed, and binary SVMs [15] are used to classify the attributes based on global features that are concatenation of local features from each CNN column.

3. EXPERIMENTS

3.1. Datasets

A-LFW. Most of the experimental results of the previous methods were reported on the LFW [16], however, the attributes labels are not available in public. We manually labeled 40 attributes for the experiment and will make it available in public. The A-LFW dataset contains 5,749 identities with totaly 13,233 images. Using the view 1 provided in [16], we split A-LFW dataset into two parts, 9,525 images for train and the remaining 3,708 images for evaluation.

CASIA-WEBFace. The CASIA-WEBFace [17] is the largest public dataset for the face recognition and provides identity labels. The dataset contain 494,414 images from 10,575 identities. We used CASIA-WEBFace dataset for pre-training of gated CNN.

3.2. Implementation Details

We extract 15 part locations using the training set of attribute-aware correlation map using training set of A-LFW dataset. The face part images were augmented by horizontal flip and were resized into 64×64. Due to the limitation in the number of A-LFW dataset, we consider two-stage learning procedure.

In the first stage, we train the gated CNN by for face recognition task to attain generalized face representation. For this stage, CASIA-WEBFace dataset with softmax and cross entropy loss is used.

In the second stage, we initialize the model with pre-trained weights. T gate layers and softmax layers are stacked on the outputs of CNN columns. The model is learned using softmax and cross entropy loss in Eqn. 2. We used Caffe [18] and libSVM [15] to implement gated CNN and SVM, respectively.

	Male	Female	Asian	Caucasian	African American	Senior	Middle-Aged	Youth	Gray Hair	Black Hair	Blond Hair	Bangs	Curly Hair	Short Hair	Straight Back Hair	Visible Forehead	Receding Hairline	Bald	Wearing Hat	Wearing Lipstick	Heavy Makeup
FaceTracer [5]	84	84	-	-	-	-	-	80	78	76	88	72	-	-	-	-	63	77	75	87	88
Deep SPN [12]	92	92	91	90	**98**	**95**	**96**	83	90	86	78	94	79	-	-	88	**91**	**96**	95	92	**97**
PANDA [11]	86	86	-	-	-	-	-	76	77	78	87	79	-	-	-	-	61	82	78	83	86
LNet+ANet [13]	94	94	-	-	-	-	-	86	84	**90**	**97**	88	-	-	-	-	85	88	88	**95**	95
Proposed w/o gate	96	96	94	89	96	87	86	90	91	87	91	94	82	85	**86**	86	81	92	**97**	**95**	93
Proposed w/ gate	**97**	**97**	**97**	**91**	**98**	86	86	**96**	**92**	88	93	**95**	**85**	**87**	**86**	**87**	84	92	**97**	**95**	93

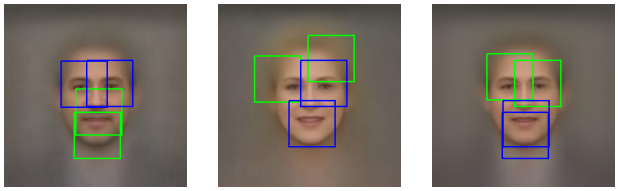
	Mustache	Beard	Sidelip Wrinkles	Forehead Wrinkles	Smiling	Frowning	Natural	Eyes Open	Eyes Wide Open	Eyes Closed	Small Eyes	Thick Eyebrows	No Eyewear	Eyeglasses	Sunglasses	Mouth Open	Mouth Wide Open	Mouth Closed	Teeth Visible		Average
FaceTracer[5]	83	69	-	-	78	-	-	-	-	-	73	67	-	90	-	77	-	-	-		79
Deep SPN [12]	**95**	**94**	-	-	91	90	-	**92**	-	**92**	-	87	90	93	98	80	87	80	**95**		90
PANDA [11]	77	63	-	-	77	-	-	-	-	-	68	63	-	84	-	74	-	-	-		78
LNet+ANet [13]	92	79	-	-	91	-	-	-	-	-	**81**	82	-	95	-	82	-	-	-		89
Proposed w/o gate	**95**	91	82	**90**	91	91	**86**	86	**84**	86	77	**86**	**98**	**97**	97	87	91	**93**	93		92
Proposed w/ gate	**95**	93	**83**	**90**	**92**	**94**	**86**	88	83	88	76	**86**	**98**	**97**	**99**	**88**	**92**	**93**	93		**93**

Table 1. Comparision result between the benchmark methods and the proposed method for 40 attribute classes on LFW dataset.

3.3. Experimental Results

Comparison with Bechmark Methods. The prediction accuracies on LFW are reported in Table 1. Due to difference of attribute classes between previously published results, we compute the averages for the common attribute classes: *male, female, youth, gray hair, black hair, blond hair, bangs, receding hairline, bald, wearing hat, wearing lipstick, heavy makeup, mustache, beard, smiling, thick eyebrows, eyeglasses*, and *mouth open*. Because FaceTracer [5] and PANDA [11] did not report the results on the LFW dataset in their original papers, we referred the results from [13]. The results of Deep SPN, and LNet+ANet are referred in the original paper. The average accuracy of the common attribute classes for FaceTracer [5], Deep SPN [12], PANDA [11], LNet+ANet [13] and the proposed method are 79%, 90%, 78%, 89%, and 93%, respectively. The experiments show that the proposed method performed best for 12 of 18 attributes and in average. The proposed method reduces 30% of the errors (from 10% to 7%) against the previous best performing method [12].

Effectiveness of Gated CNNs. In Table 1, the comparison results between gated CNN with a gate layer and the CNN without gate layer is described. As depicted in Figure 3.3, the gate layer prevents the CNNs to be over influenced by irrelevant attribute labels, and it leads to performance improvement. For 24 of 40 attribute classes, the accuracy is improved by using gate layers, and for other attribute the accuracies are similar or decreased 1% in maximum.



(a) Beard (b) Blond hair (c) Eyes open

Fig. 3. The mean faces of (a) *Beard*, (b) *Blond hair*, and (c) *Eyes open* are overlaid with four part regions that are inputted to gated CNN. For better visualization, two face regions corresponding to the highest gate weights in the gate layer (green), and two face regions corresponding to the lowest gate weights in the gate layer (blue) are selected.

4. CONCLUSION

In this paper, a face attribute classification method that uses attribute-aware correlation map and gated convolutional neural networks (CNN) is proposed. The attribute-aware correlation map considers pixel-wise correlations between face images and attribute labels for providing more precise face part regions that are correlated with attributes. Based on the face part regions, gated CNN simultaneously learns face representations and selects corresponding face part regions by neglecting uncorrelated attribute labels with face part regions.

The experiments on the LFW dataset with 40 attribute classes show significant performance improvement.

5. REFERENCES

[1] Neeraj Kumar, Peter Belhumeur, and Shree Nayar, "Facetracer: A search engine for large collections of images with faces," in *Computer Vision–ECCV 2008*, pp. 340–353. Springer, 2008.

[2] Behjat Siddiquie, Rogério Schmidt Feris, and Larry S Davis, "Image ranking and retrieval based on multi-attribute queries," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 801–808.

[3] Devi Parikh and Kristen Grauman, "Relative attributes," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 503–510.

[4] Junyoung Chung, Donghoon Lee, Youngjoo Seo, and Chang D Yoo, "Deep attribute networks," *arXiv preprint arXiv:1211.2881*, 2012.

[5] Neeraj Kumar, Alexander C Berg, Peter N Belhumeur, and Shree K Nayar, "Describable visual attributes for face verification and image search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 10, pp. 1962–1977, 2011.

[6] Corinna Cortes and Vladimir Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[7] Lubomir Bourdev, Subhransu Maji, and Jitendra Malik, "Describing people: A poselet-based approach to attribute classification," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1543–1550.

[8] Lubomir Bourdev and Jitendra Malik, "Poselets: Body part detectors trained using 3d human pose annotations," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1365–1372.

[9] Thomas Berg and Peter N Belhumeur, "Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 955–962.

[10] Peter N Belhumeur, David W Jacobs, David Kriegman, and Neeraj Kumar, "Localizing parts of faces using a consensus of exemplars," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 545–552.

[11] Ning Zhang, Manohar Paluri, Marc'Aurelio Ranzato, Trevor Darrell, and Lubomir Bourdev, "Panda: Pose aligned networks for deep attribute modeling," in *Computer Vision and Pattern Recognition*, 2014.

[12] Ping Luo, Xiaogang Wang, and Xiaoou Tang, "A deep sum-product architecture for robust facial attributes analysis," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2864–2871.

[13] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang, "Deep learning face attributes in the wild," *arXiv preprint arXiv:1411.7766*, 2014.

[14] Vahid Kazemi and Josephine Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 1867–1874.

[15] Chih-Chung Chang and Chih-Jen Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, pp. 27, 2011.

[16] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Tech. Rep. 07-49, University of Massachusetts, Amherst, October 2007.

[17] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li, "Learning face representation from scratch," *arXiv preprint arXiv:1411.7923*, 2014.

[18] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2014, pp. 675–678.