Harnessing Synthesized Abstraction Images to Improve Facial Attribute Recognition

Keke He[1*], Yanwei Fu[2*], Wuhao Zhang[4], Chengjie Wang[3], Yu-Gang Jiang[1†], Feiyue Huang[3], Xiangyang Xue[1]

[1]School of Computer Science, Shanghai Key Lab of Intelligent Information Processing, Fudan University
[2]School of Data Science, Fudan University
[3]Tencent Youtu Lab
[4]Shanghai Jiao Tong University
{kkhe15, yanweifu}@fudan.edu.cn, vv_xiaod@sjtu.edu.cn, jasoncjwang@tencent.com, ygj@fudan.edu.cn, garyhuang@tencent.com, xyxue@fudan.edu.cn

Abstract

Facial attribute recognition is an important and yet challenging research topic. Different from most previous approaches which predict attributes only based on the whole images, this paper leverages facial parts locations for better attribute prediction. A facial abstraction image which contains both local facial parts and facial texture information is introduced. This abstraction image is generated by a Generative Adversarial Network (GAN). Then we build a dual-path facial attribute recognition network to utilize features from the original face images and facial abstraction images. Empirically, the features of facial abstraction images are complementary to features of original face images. With the facial parts localized by the abstraction images, our method improves facial attributes recognition, especially the attributes located on small face regions. Extensive evaluations conducted on CelebA and LFWA benchmark datasets show that state-of-the-art performance is achieved.

1 Introduction

Facial attribute recognition has received extensive research attention over the past decades. Facial attributes are used to describe the person characteristics of a face image. Learning to predict facial attributes can not only be used as the intermediate representations for other learning tasks such as face recognition [Wang *et al.*, 2017b; Hu *et al.*, 2017], but also directly useful for real-world applications such as face retrieval [Siddiquie *et al.*, 2011], and intelligent retail. For example, analyzing facial attributes can automatically detect the age and gender of customers in the shopping malls and thus helps these commercial agents to accumulate and understand the Big Data of customer styles.

Learning a robust model for facial attribute recognition is very challenging primarily due to the difficulties of parsing input face images. Specifically, the input face images may contain very noisy and dynamic background, e.g., the scene of a shopping mall. This background information may negatively affect the recognition process of facial attributes. Furthermore, most types of facial attributes (e.g., eyeglasses, or arched eyebrows) can be localized to some particular regions of faces. For example, the "wearing hat" attribute is mostly corresponding to the hair part of human faces without needing the information from other parts of the image, *say*, the mouth. Isolating the local regions to learn each type of attribute can help facial attribute recognition.

To directly parse the local parts of faces, previous works either use the landmarks to crop face region by bounding box [Kumar *et al.*, 2009], or directly segment the face images into facial parts [Kalayeh *et al.*, 2017]. The former methods may include the undesirable parts. For instance, if using the bounding box to crop the hair part, it may crop the whole face region if the person has long hair. The latter one may result in losing the details of texture information. This detailed information is nevertheless very critical for facial attribute recognition. In contrast, this paper "isolates" the important factor to predict the facial attributes with the facial abstraction task. We aim at generating abstracted facial regions from original face images that is possible to remove the useless background but still contains the facial part locations information.

The facial abstraction task is inspired by the task of facial segmentation which parses the face images into meaningful facial parts. The key difference is that our facial abstraction task will require the parsing algorithm to save as much texture information from the original images as possible. Essentially, facial abstraction process can be implemented by the recent Generative Adversarial Network (GAN) model [Goodfellow *et al.*, 2014]. After obtaining the synthesized facial abstraction image, the original image and abstracted image are fed into a dual-path network which contains original image subnet and abstraction image subnet. To further leverage the information from the abstraction sub-

*This work was done when Keke He was an intern at Tencent Youtu Lab. The first two authors contributed equally to this paper.

†Yu-Gang Jiang is the corresponding author.

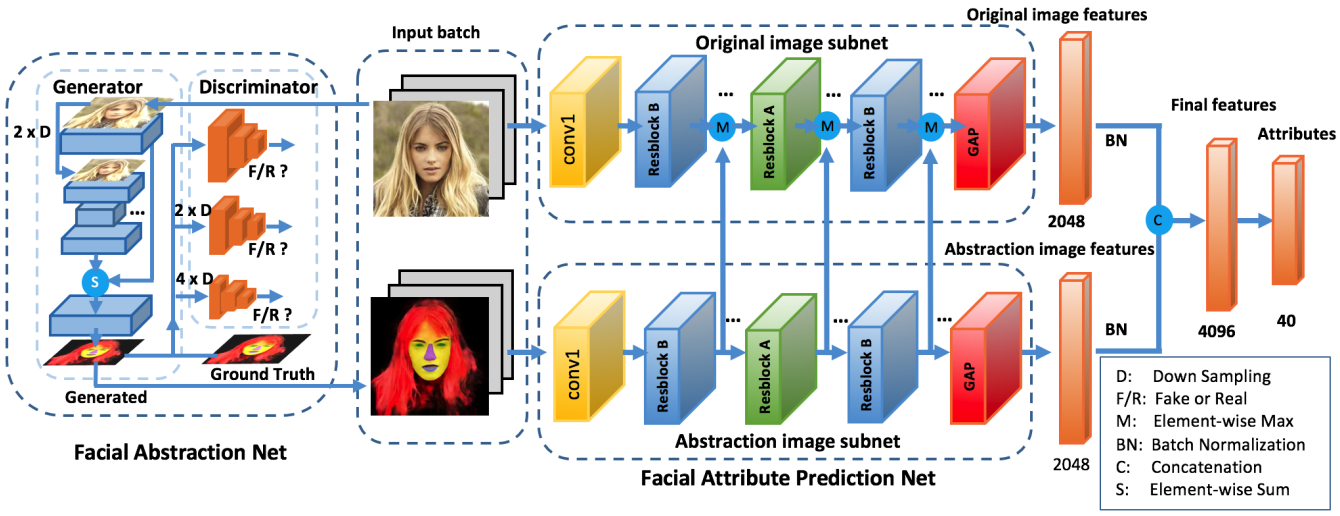

Figure 1: Overview of the proposed architecture. The facial abstraction net is based on pix2pixHD.

net, the feature maps of the abstraction subnet are passed to original image subnet. Finally, these two features are concatenated for final attribute recognition. Our attribute recognition network is trained in an end-to-end manner. We evaluate our proposed framework on benchmarks including CelebA [Liu *et al.*, 2015b], LFWA [Huang *et al.*, 2007; Liu *et al.*, 2015b] face attribute datasets and the experiment results significantly outperform the state-of-the-art alternatives.

In summary, our main contribution is to propose a systematic way of harnessing synthesized abstraction images to help improve facial attribute recognition. In particular, (1) To the best of our knowledge, we for the first time utilize the GAN model to generate the facial abstraction images which contain the part locations and textual information. (2) We are the first to propose a dual-path network to combine the synthesized abstraction images and original images to help attribute recognition. (3) We show that attribute recognition can be improved with the help of abstraction images. We evaluate the framework on two benchmark datasets, and the experimental results validate the effectiveness of our method.

2 Related Work

Facial Attribute Recognition. In term of distinctive learning paradigm, the facial attribute recognition can be divided into two categories: part-based and holistic approaches. For the part-based method, it contains an attribute-related part detector and then extracts features on the localized facial parts. [Kumar *et al.*, 2009] employed hand-crafted features to parse pre-defined facial parts to facilitate training SVM for facial attribute recognition. [Zhang *et al.*, 2014] employed poselets [Bourdev *et al.*, 2011] to detect body parts to extract Convolutional Neural Network (CNN) features of the localized parts.

On the other hand, various deep multi-task architectures [Liu *et al.*, 2015b; Rudd *et al.*, 2016; Lu *et al.*, 2017; Han *et al.*, 2017] are holistically learned for facial attribute recognition. Comparing with all these previous methods, the GAN model is learned in our framework to parse facial parts to better help attribute prediction. [Ding *et al.*, 2017] designed a weakly-supervised face region aware network to automatically detect face regions, while ours learns a GAN to obtain parts locations.

[Kalayeh *et al.*, 2017] employed the semantic segmentation to improve facial attribute prediction; in contrast, we utilize synthesized abstraction images. Specifically, (1) Different frameworks to generate segmentation/abstraction: [Kalayeh *et al.*, 2017] adopted an encoder-decoder in generating the segmentation, rather than the synthesized abstraction images produced by GAN in our framework. (2) Different ways of using segmentation/abstraction for prediction. The segmentation images are used in [Kalayeh *et al.*, 2017] as masks to pool/gate the activations (features) for prediction. In contrast, our synthesized abstraction images are directly used to train a network for attribute prediction. Critically, the network trained by synthesized images is able to achieve relative competitive results compared with the other baselines as shown in Tab. 1, Tab. 3, Tab. 4 and Tab. 5.

Face Segmentation and Face Inpainting. Face segmentation is also called as face parsing. It gives a semantic class label to every pixel in a face image, results in segmenting the input face image into semantic regions, e.g, hair, eyes and nose for further analysis. Researchers have developed several face segmentation methods based on Conditional Random Field (CRF), exemplar [Smith *et al.*, 2013] and deep neural network [Liu *et al.*, 2015a]. For the exemplar-based methods, [Smith *et al.*, 2013] proposed a method based on transferring labeling masks from registered exemplars images to the test image in a pixel-wise manner. For the deep neural network based methods, [Luo *et al.*, 2012] developed a deep parsing framework based on deep hierarchical features and separately trained models. [Liu *et al.*, 2015a] proposed a multi-objective deep network that can jointly learn pixel-wise likelihoods and pairwise label dependencies. Similarly, face inpainting refers to the technique of modifying a face

image with partial occlusions due to sunglasses and hand in a seamless manner. An early attempt to inpainting is by [Mo *et al.*,], which reconstructed the occluded region of a face by a linear combination of several face images. Recently, [Jampour *et al.*, 2017] introduced a data-driven approach which made use of inferred high-level facial attributes, such as gender, ethnicity, and expression. There are some methods which use generative model to inpainting [Pathak *et al.*, 2016; Yeh *et al.*, 2017]. [Yeh *et al.*, 2017] proposed a method that learned to generate the missing content by searching for the closest encoding of the corrupted image in the latent image manifold. Different from face segmentation and face inpainting tasks, our facial abstraction task not only generates the facial parts but also contains an amount of textual information. The generated results are basically learned and abstracted from a large amount of training data. Thus the abstracted image results are not only based on the input image but also get affected by those images that are mostly similar to the input image.

3 Methodology

We propose a dual-path deep convolutional neural network for facial attribute recognition. The framework is illustrated in Fig. 1. It is composed of a facial abstraction network and a facial attribute prediction network. The facial attribute prediction network is composed of two subnets. The features of two subnets are concatenated after batch normalization [Ioffe and Szegedy, 2015]. The concatenated features are used for the final attribute recognition by a sigmoid cross entropy loss layer. Each component will be described next, including the face attribute recognition problem in Sec. 3.1 and the structure of base attribute recognition network in Sec. 3.2. Then we will introduce the way to generate abstraction images in Sec. 3.3. Finally, the training process will be discussed in Sec. 3.4.

3.1 Problem Setup

We aim to learn the attribute classifiers that can predict the existence of attributes of face images. Assume we have the training dataset $\mathcal{D}_s = \{\mathbf{I}, \mathbf{a}, \mathbf{L}\}$ with N training images and M attributes. $\mathbf{I}$ denotes the training instances, $\mathbf{a}$ is the attribute names and $\mathbf{L}$ denotes the labels. If the i-th image $\mathbf{I}_i$, $(i = 1, \cdots, N)$ is annotated to have the $j-$th attribute $\mathbf{a}_j$ $(j = 1, \cdots, M)$, we denote $\mathbf{L}_{ij} = 1$; otherwise, $\mathbf{L}_{ij} = 0$. Given a unseen test image $\mathbf{I}^\star$, the goal is then to learn a mapping function $\mathbf{a}^\star = \Psi(\mathbf{I}^\star)$ using all available training information and predict the attribute vector $\mathbf{a}^\star$. As each image can be labelled with multiple attributes, the predicting functions can be written as $\Psi = [\psi_j]_{j=1,\cdots,M}$, and $\psi_j(\mathbf{I}^\star) \in \{+1, 0\}$.

3.2 Basic Attribute Prediction Network

Our basic attribute prediction network is illustrated in Fig. 2. It includes the convolutional layers, pooling layers, fully connected layers and residual block layers [He *et al.*, 2015].

Convolutional Layers. This type of layer pre-processes the input image for the following steps. In particular, the first convolutional layer is set as 7×7 kernel size in order to guarantee a large receptive field. For all the other convolutional layers, the kernel size is 3×3. Except for the first convolutional layer, all the other convolutional layers are used to construct two types of residual blocks – Resblock A and Resblock B.

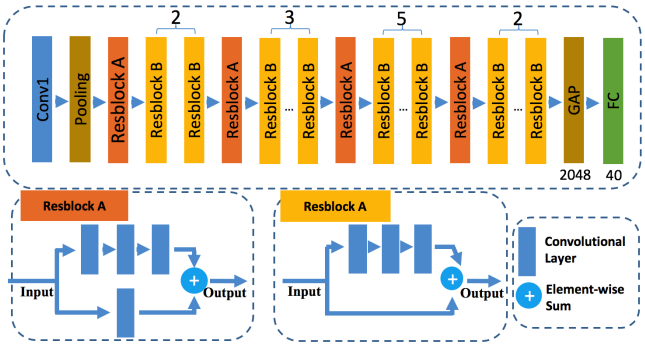

Figure 2: The structure of our basic attribute prediction network. It is based on ResNet50. Note that: GAP represents the global average pooling layer.

Residual Block Layers. For all the residual blocks, it has 3 convolutional filters as the main road. (1) In ResBlock A, there is one convolutional filter on the side road. (2) In ResBlock B, there is a bypass directly to the output. Finally, these two roads are connected by an element-wise sum operation. After the final residual block, Global Average Pooling (GAP) layer is applied to produce a 2048-D vector representation.

Fully Connected Layers. This layer converts 2048-D features to M attribute values, where M is the number of attributes. This basic structure is used to construct two subnets in our dual-path attribute prediction model. In the previous methods, the Euclidean loss is used as the loss function [Rudd *et al.*, 2016]. In contrast, we train the network using sigmoid cross entropy loss, which is shown better at predicting the facial attributes in the experiments.

3.3 Facial Abstraction Network

The facial abstraction network aims at synthesizing an abstraction of the image from the original image. The recent GAN-based method is applied for such purpose. Essentially, GAN has two components: the generator and the discriminator. The generator aims at learning to synthesize images that are indistinguishable from the natural images, while the discriminator is optimized to differentiate the synthesized images from the real natural images. In particular, we utilize the pix2pixHD [Wang *et al.*, 2017a] to learn to generate the facial abstraction image. It has two components: a coarse-to-fine generator G and a multi-scale discriminator D.

The pix2pixHD tries to produce a realistic natural image by a given segmentation image. In contrast, as shown in Fig. 1, our method takes the natural images as the input and generates the abstracted face images. Our training data is a set of pairs of images (r_i, a_i), where r_i is the real images and a_i is the corresponding abstraction images. Our GAN aims to model the conditional distribution of abstraction images

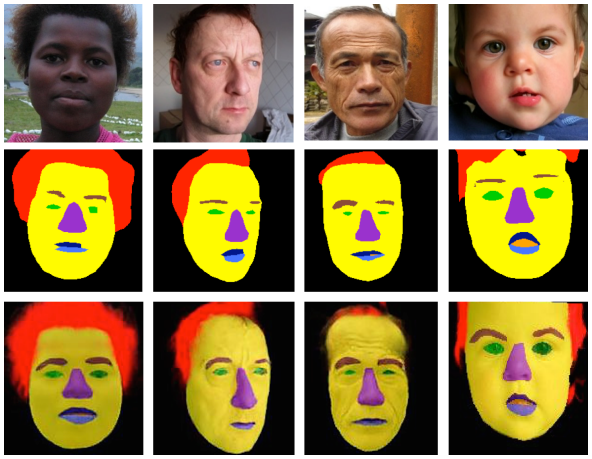

Figure 3: Comparing the face segmentation and abstraction results on Helen testing set. The first row is the original images. The second row is the segmentation result generated by Deeplabv2; images are colored by the 11 ground truth labels. The third row is our facial abstraction result.

given the real images by the following objective function,

$$\min_G \max_D \mathcal{L}_{GAN(G,D)} = \mathbb{E}_{r,a\sim p_{data}(r,a)}\left[\log D\left(r,a\right)\right] + \mathbb{E}_{r\sim p_{data}(r)}\left[\log\left(1-D\left(r,G\left(r\right)\right)\right)\right], \quad (1)$$

In particular, the pix2pixHD used 3 discriminators (D_1, D_2 and D_3) corresponding to three scales of images. The down-sampling operators are conducted on the real and synthesized images by a factor of 2 and 4 respectively, in order to get the images used for D_2 and D_3 respectively. Thus we can formulate learning GAN as a multi-task learning problem as,

$$\min_G \max_{D_1,D_2,D_3} \sum_{k=1}^{3} \mathcal{L}_{GAN(G,D_k)} \quad (2)$$

Our facial abstraction images are also compared against the results of face segmentation produced by Deeplabv2 [Chen *et al.*, 2016]. The visualization results are shown in Fig. 3. The first, second and third rows are the images of original, face segmentation and facial abstraction respectively. Apparently, the facial abstraction images have saved more texture information than segmentation images. The details of training procedure will be described in 4.1.

3.4 Training Process

In this section, we will describe the training process which leverages information from the facial abstraction images. The original face images and abstraction images are paired as the input to the network, these two kinds of images go to the two subnets: original image subnet and abstraction image subnet respectively. As these two images have a different visual appearance, to fully explore the information in the two input images, the weights of two subnets are unshared. Global average pooling is applied after the last residual block layer and then the features from two subnets are obtained. Additional batch normalization layer is added to normalize two types of features. The normalized features are concatenated to generate the final features to predict attributes. In order to better leverage the abstraction image information, the connections from the abstraction subnet to the original subnet are added. Specially, after each residual block, the output feature maps of the abstraction image are passed to the original subnet. These two outputs are fused by an element-wise max operation.

4 Experiments

4.1 Datasets and Experimental Settings

We conduct experiments on two most widely used datasets. **(1) CelebA** contains 202,599 images of approximately 10k identities [Liu *et al.*, 2015b]. Each image is annotated with 40 binary attributes. For a fair comparison with the other methods, we follow the standard split here: the first $162,770$ images are used for training, $19,867$ images for validation and remaining $19,962$ for testing. CelebA provides the pre-cropped face images and we use cropped images to train and test attribute models same as the other methods [Rudd *et al.*, 2016]. **(2) LFWA** [Liu *et al.*, 2015b] is constructed based on face recognition dataset LFW [Huang *et al.*, 2007]. LFWA has a total of 13,232 images of 5,749 identities with predefined train and test splits which divide the entire dataset into two approximately equal partitions. We follow the partition of data to train and test our model. In LFWA, each image has 40 binary facial attributes, the same as CelebA.

Evaluation Metrics. The facial attribution recognition can be taken as the problem of classification tasks. To evaluate the performance, (1) mean accuracy (*acc*) over all attributes is computed. This metric has also been used in previous work [Liu *et al.*, 2015b]. (2) Further, we find the positive and negatives instances per attribute are extremely imbalanced in the CelebA dataset. For example, for the "Bald" attribute, we can get a high accuracy of 97.88% if predicting all the test images have no bald. To appropriately evaluate the quality of different methods, following the evaluation metrics used in pedestrian attribute recognition problem [Li *et al.*, 2016], we add four more evaluation metrics, a label-based metric mean balanced-accuracy, short in *bal-acc*, and three instance-based metrics, *i.e.* precision (*prec*), recall (*rec*) and F1-score (*F1*). Formally, the *acc* and *bal-acc* can be calculated as,

$$acc = \frac{1}{M}\sum_{i=1}^{M}(T_i/N) \quad (3)$$

$$bal-acc = \frac{1}{2M}\sum_{i=1}^{M}(TP_i/P_i + TN_i/N_i) \quad (4)$$

where M is the total number of attributes; N and T_i are the numbers of examples and correctly predicted examples; P_i and TP_i are the numbers of positive examples and correctly predicted positive examples; N_i and TN_i are the numbers of negative examples and correctly predicted negative examples.

Parameter Settings. We use the open source deep learning framework Caffe [Jia *et al.*, 2014] to train our network. For all the experiments, we only use a single end-to-end model for testing. We use the stochastic gradient descent algorithm to train our models. (1) On CelebA dataset, the weights of convolutional layers are initialized by the ResNet50 [He *et*

Methods	Accuracy (%)
[Kumar *et al.*, 2008] FaceTracer	81.12
[Zhang *et al.*, 2014] PANDA-w	79.00
[Zhang *et al.*, 2014] PANDA-l	85.00
[Liu *et al.*, 2015b] LNets+ANet	87.30
[Ehrlich *et al.*, 2016] MT-RBM-PCA	87.00
[Zhong *et al.*, 2016] Off-the-Shelf CNN	86.60
[Wang *et al.*, 2016] Walk-and-Learn	88.00
[Rudd *et al.*, 2016] Rudd et al. Separate	90.22
[Rudd *et al.*, 2016] Rudd et al. Moon	90.94
[Lu *et al.*, 2017] SOMP-branch-32	90.74
[Hand and Chellappa, 2017] MCNN-AUX	91.26
[Ding *et al.*, 2017] PaW	91.23
[Kalayeh *et al.*, 2017] Avg. Pooling	90.86
[Kalayeh *et al.*, 2017] SSG	91.62
[Kalayeh *et al.*, 2017] SSP	91.67
[Kalayeh *et al.*, 2017] SSP+SSG	91.80
Original	91.50
Abstraction	90.36
Ours	**91.81**

Table 1: Comparison of mean accuracy on CelebA with state-of-the-art methods.

Attributes	Original Images(%)	Our Model (%)
ArchedEyebrows	84.28	**85.15**
BagsUnderEyes	85.27	**85.77**
BushyEyebrows	92.77	**93.07**
Eyeglasses	99.68	**99.72**
NarrowEyes	**87.85**	87.81

Table 2: Comparison of eye/eyebrow related attributes on CelebA with baseline model.

Methods	Accuracy (%)
[Kumar *et al.*, 2008] FaceTracer	73.93
[Zhang *et al.*, 2014] PANDA-w	79.00
[Zhang *et al.*, 2014] PANDA-l	81.00
[Liu *et al.*, 2015b] LNets+ANet(w/o)	79.00
[Liu *et al.*, 2015b] LNets+ANet	84.00
[Kalayeh *et al.*, 2017] Avg. Pooling	85.27
[Kalayeh *et al.*, 2017] SSG	86.13
[Kalayeh *et al.*, 2017] SSP	86.80
[Kalayeh *et al.*, 2017] SSG+SSP	**87.13**
ResNet18 + SVM	82.35
ResNet50 + SVM	83.09
Original	84.79
Abstraction	83.64
Ours	85.28

Table 3: Comparison of mean accuracy on LFWA datasets with state-of-the-art methods.

al., 2015] network that is pre-trained on ImageNet dataset. The base learning rate is set as 0.001 and gradually decreased by $1/10$ at $20k$, $45k$ iterations. The input image is resized to 224×224. (2) On LFWA dataset, due to the relatively small number of training samples ($6k$), we adopt a smaller network structure ResNet18 [He *et al.*, 2015] to avoid overfitting. The base learning rate is still set as 0.001 and gradually decreased by $1/10$ at $1k$, $2k$ iterations.

Running Costs. Our model trained on CelebA dataset gets converged with $46k$ iterations and it takes 10 hours with one NVIDIA Tesla M40 GPU. Our model trained on LFWA gets converged with $2.5k$ iterations and it takes half an hour. For training all the model, the batch size is 20, and it takes around 22 GB GPU memory.

Facial Abstraction Networks. This network is trained by the Helen dataset, which is a widely used dataset for face parsing. [Le *et al.*, 2012; Smith *et al.*, 2013]. In this dataset, each image is annotated with 11 segment classes. These labels are as follows: background, face skin (excluding ears and neck), left eyebrow, right eyebrow, left eye, right eye, nose, upper lip, inner mouth, lower lip, and hair. It is composed of total $2,330$ images and divided into $2,000$ training images, 230 validation images, and 100 testing images. We train face abstraction model on the training images. To generate the ground truth abstraction images, we use the codes of [Liu *et al.*, 2015a] which saves the textual information. To train the facial abstraction model, we use the codes of [Wang *et al.*, 2017a]. As our GAN learns the distribution of training data (including textual information), we can generate images with textual information. Our input image and the label image are resized to 256×256. We use Adam with the learning rate of 0.0002 to optimize our abstraction network. The batch size is 1. We train the network with 200 epochs. It takes 37 hours with one NVIDIA Tesla M40 GPU and needs around 13 GB GPU memory. We then apply the abstraction network to the face attribute recognition datasets. Even very few numbers of Helen training data used in our training process, the abstraction model is able to color various facial regions successfully in unseen images. Later, we evaluate our proposed attribute prediction model where these abstraction cues are utilized to improve facial attribute recognition.

4.2 Competitors

We compare our results against state-of-the-art methods and baselines. Particularly, (1) **FaceTracer** [Kumar *et al.*, 2008] extracts the HOG and color histograms in manually defined facial parts and then trains SVM for each attribute recognition. (2) **PANDA** [Zhang *et al.*, 2014] uses poselets [Bourdev *et al.*, 2011] to detect parts and then extracts CNN features from the localized parts. (3) **LNets+ANet** [Liu *et al.*, 2015b] employs two deep CNNs to localize face and one deep CNN network to learn facial feature. (4) **Off-the-Shelf CNN** [Zhong *et al.*, 2016] extracts features from the off-the-shelf face recognition model. (5) **Walk and Learn** [Wang *et al.*, 2016] exploits videos and contextual data to learn representations for facial attributes. (6) **Moon** [Rudd *et al.*, 2016] learns a mixed objective optimization network for learning each attribute and utilizes distribution of attribute labels. (7) **SOMP** [Lu *et al.*, 2017] learns a deep multi-task learning framework which can dynamically group similar tasks together. (8) **MCNN-AUX** [Hand and Chellappa, 2017] takes the attribute

Methods	Acc.	Bal-acc.	Prec.	Rec.	F1
Original	91.50	81.57	**84.67**	76.20	80.21
Abstraction	90.36	78.64	82.11	73.66	77.66
Our Model	**91.81**	**83.13**	84.22	**78.21**	**81.10**

Table 4: Comparison with the baseline models on CelebA.

Methods	Acc.	Bal-acc.	Prec.	Rec.	F1
Original	84.79	76.48	80.90	74.19	77.40
Abstraction	83.64	75.05	79.09	72.52	75.67
Our Model	**85.28**	**77.50**	**82.01**	**74.22**	**77.92**

Table 5: Comparison with the baseline models on LFWA.

relationships into consideration for improving classification accuracy. (9) **PaW** [Ding *et al.*, 2017] combines multiple part-based networks and a whole-image-based network for final attribute classification. (10) [Kalayeh *et al.*, 2017] learns an encoder-decoder to produce the segmentation images, and then leverages the segmentations as masks to pool/gate the activations for attribute prediction. Here we compare three different variants of [Kalayeh *et al.*, 2017] — **Average Pooling, SSG, SSP**. These three variants configure three different ways of utilizing the segmented images to pool/gate the feature maps and thus help facial attribute recognition. (11) **Original** is one variant of our model. Original images are used as input to train the model. (12) **Abstraction** is another variant of our model. It uses the abstraction image as the input. (13) **ResNet18 + SVM** is one baseline model. It extracts the features from a whole face image by a ResNet18 model which pre-trained on ImageNet2012, and then trains one SVM classifier for each attribute. (14) **ResNet50 + SVM** is another baseline model. Features from ResNet50 model pre-trained on ImageNet2012 are used to train separate SVM for each attribute.

4.3 Results on CelebA Dataset

We evaluate the facial attribute recognition task with the standard settings of CelebA dataset. The results are listed in Tab. 1. We highlight the following observations.

(1) State-of-the-art results. The results of our model beat all the state-of-the-art methods. Comparing with all the other methods, we highlight that our method achieves the best performance with the mean accuracy of 91.81% over 40 facial attributes. The results show 5.21%, 3.81% and 0.89% improvement over Off-the-Shelf [Zhong *et al.*, 2016], Walk-and-Learn [Wang *et al.*, 2016], Moon [Rudd *et al.*, 2016] respectively. In particular, comparing with the current state-of-the-art method LNets+ANet [Liu *et al.*, 2015b] which has a classification error of 12.70%, our method with an error of 8.19%, reducing the classification error by 35.5%. This improved performance validates the effectiveness of our framework. It is important to note that, [Wang *et al.*, 2016] used 5 million auxiliary image pairs to pre-train their model, and [Lu *et al.*, 2017] employed the face recognition model as the pre-train model.

(2) Effectiveness of facial abstraction subnet. We compare the other variants of our model and show the efficacy of abstraction subnet. Specifically, we compare several baseline models: Original and Abstraction, we find that even training with the abstraction images, our abstraction baseline model can get a mean accuracy of 90.36%, which can beat the most of the state-of-the-art methods. This validates our abstraction images can well represent the original images, preserving the detailed facial information. Besides, our dual-path model can obtain a mean accuracy of 91.81%, and it shows 1.45%, 0.31% improvement over the abstraction and original model baselines individually. This is due to the fact that the features of original image and abstraction image are complementary to each other. And more critically, our dual-path network can efficiently combine them to produce very competitive results.

(3) Finally, we compare our results with [Kalayeh *et al.*, 2017]. In particular, (1) We highlight that this is the first work of utilizing synthesized images to help facial attribute prediction. The synthesized images are capable of training a network in attribute prediction. We further harness these synthesized images to improve the performance. (2) Both methods are very good, and yet we are using different strategies in the way of generating segmentation/abstraction images and using segmentation/abstraction for prediction. Compared with the mean accuracy, our results are very marginally better than [Kalayeh *et al.*, 2017] on CelebA dataset, and slightly worse on LFWA dataset. Note that the CelebA dataset which has of $162k$ and $20k$ images for training and testing individually is much larger than the LFWA dataset of 6k training and 7k testing images. This shows that our methods are comparable to the state-of-the-art methods in [Kalayeh *et al.*, 2017].

4.4 Results on LFWA Dataset

To further test the proposed method, we applied it to the LFWA face attribute dataset. We find that (1) Again the results of our model are better than or have comparable performance to the state-of-the-art methods. As we can see from Tab. 3, ours achieve the mean accuracy of 85.28% over 40 facial attributes. In particular, it shows 1.38% improvement over the current state-of-the-art LNets+ANet [Liu *et al.*, 2015b]. This validates the effectiveness of the proposed attribute classification network. (2) Furthermore, this experiment still validates the efficacy of parsing subnet. We list the result of the baseline models in Tab. 5. Our model can obtain the mean accuracy of 85.28%, and it shows 1.59% and 0.49% improvement over two baseline models: abstraction images and original images respectively. This validates the efficacy of the abstraction image features. It is complementary to original images features. Meanwhile, the abstraction image can help to aware the locations of different facial components, thus improving the attribute recognition accuracy. Our method also shows 2.93% and 2.19% improvement over two SVM baseline models.

4.5 More Evaluation Metrics

We further compare our results with the baselines on more metrics. In particular, for the significant imbalance classification task, especially the facial attribute recognition, mean classification accuracy is not the best evaluation metric. Thus extensive study by using different evaluation metrics has been conducted and compared in Tab. 4 and Tab. 5. These metrics

Methods	Acc.	Bal-acc.	Prec.	Rec.	F1
W./O. norm	91.53	81.59	**84.44**	76.53	80.29
W. norm	**91.81**	**83.13**	84.22	**78.21**	**81.10**

Table 6: Comparison of with / without feature normalization on CelebA. W./O. norm and W. norm represent the methods without and with normalization.

Methods	Acc.	Bal-acc.	Prec.	Rec.	F1
Euc. Loss	**91.51**	79.36	**86.90**	73.45	79.61
S.C.E. Loss	91.50	**81.57**	84.67	**76.20**	**80.21**

Table 7: Comparison of different loss on CelebA. Euc. Loss and S.C.E. Loss indicate the Euclidean loss and Sigmoid Cross Entropy Loss respectively.

definitions are the same as those in pedestrian attribute recognition [Deng *et al.*, 2014], including a label-based metric mean balanced accuracy (*bal-acc*) and three instance-based metrics precision (*prec*), recall (*rec*) and F1-score (*F1*). These metrics can systematically evaluate the performance of our methods over baselines.

For example, on CelebA dataset, our dual-path model can achieve the 77.50 *bal-acc*, which outperforms the abstraction and original baselines by 2.45 and 1.02 respectively. Furthermore, our dual-path model hits the 81.10 *F1*, which improves over the abstraction model and original models by 3.44 and 0.89 individually. On LFWA dataset, we report our *F1* results of 77.92, which beats the two baselines again. Thus overall our results are still better than the baseline models.

5 Ablation Study

Analysis of attributes on small face regions. To further evaluate the abstraction subnet, we select the attributes which related to eye or eyebrows on CelebA dataset. In a face image, these two face components always occupy limited regions. We list the accuracy results on Tab. 2. Comparing our model with baseline original image model, our model has improvement on all the attributes except the NarrowEyes attribute. This may reveal that with the help of abstraction images, our model can aware the small but important parts of facial images, thus improving the accuracy of these attributes.

The choice of the loss function. We evaluate the loss function for binary attribute prediction network. [Rudd *et al.*, 2016] uses the Euclidean loss to regress attribute labels, [Kalayeh *et al.*, 2017] uses the sigmoid cross entropy loss to classify attributes. To evaluate which loss is better, we apply different loss on the CelebA dataset, the results are listed in Tab. 7. If compared with the mean accuracy metric, these two losses can achieve comparable results with 91.51% and 91.50% respectively. We further evaluate this two loss functions on mean balanced-accuracy, precision, recall and F1 metrics. Sigmoid cross entropy loss has 2.21 improvement on mean balanced-accuracy and 0.60 improvement on F1. Euclidean loss can only beat cross entropy loss on the precision metric. This reveals sigmoid cross entropy loss is better for binary attribute classification. Thus, we adopt sigmoid cross entropy loss to train all attribute models.

The importance of feature normalization. This study evaluates the importance of feature normalization. In our model, after the last pooling layer, the features of the face image and abstraction image are obtained. Before the feature concatenation, we compare our framework with feature normalization and without feature normalization. To perform feature normalization, we add additional batch normalization layer after the last pooling layer. The results are listed in Tab. 6. As we can see from the table, with feature normalization method can achieve 0.28% mean accuracy and 1.54% mean balanced-accuracy improvement over without feature normalization method. This reveals feature normalization is important before concatenation.

6 Conclusion

In this paper, we propose a novel dual-path convolutional neural network to learn facial attributes. Different from most previous approaches which predict attributes only based on the whole images, our method utilizes synthesized facial abstraction images to help attribute recognition tasks. The proposed framework fuses the features from original images and facial abstraction images to learn all the attributes tasks. We demonstrate our approach on the CelebA, LFWA attribute datasets, showing substantial improvement over the state-of-the-art methods.

Acknowledgments

The authors would like to thank anonymous reviewers for their helpful comments. The authors are also grateful for valuable suggestions from Ying Tai and Yanhao Ge. This work was supported in part by National Key R&D Program of China (No.2017YFC0803700), NSFC under Grant (No.61572138 & No.U1611461) and STCSM Project under Grant (No.16JC1420400 & No.2017SHZDZX01).

References

[Bourdev *et al.*, 2011] Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Describing people: A poselet-based approach to attribute classification. In *ICCV*. IEEE, 2011.

[Chen *et al.*, 2016] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *arXiv:1606.00915*, 2016.

[Deng *et al.*, 2014] Yubin Deng, Ping Luo, Chen Change Loy, and Xiaoou Tang. Pedestrian attribute recognition at far distance. In *ACM MM*, 2014.

[Ding *et al.*, 2017] Hui Ding, Hao Zhou, Shaohua Kevin Zhou, and Rama Chellappa. A deep cascade network for unaligned face attribute classification. In *AAAI*, 2017.

[Ehrlich *et al.*, 2016] Max Ehrlich, Timothy J Shields, Timur Almaev, and Mohamed R Amer. Facial attributes classification using multi-task representation learning. In *CVPR Workshops*, 2016.

[Goodfellow *et al.*, 2014] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley,

Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[Han *et al.*, 2017] Hu Han, Anil K Jain, Shiguang Shan, and Xilin Chen. Heterogeneous face attribute estimation: A deep multi-task learning approach. *TPAMI*, 2017.

[Hand and Chellappa, 2017] Emily M Hand and Rama Chellappa. Attributes for improved attributes: A multi-task network utilizing implicit and explicit relationships for facial attribute classification. In *AAAI*, 2017.

[He *et al.*, 2015] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv:1512.03385*, 2015.

[Hu *et al.*, 2017] Guosheng Hu, Yang Hua, Yang Yuan, Zhihong Zhang, Zheng Lu, Sankha S Mukherjee, Timothy M Hospedales, Neil M Robertson, and Yongxin Yang. Attribute-enhanced face recognition with neural tensor fusion networks. In *ICCV*, 2017.

[Huang *et al.*, 2007] 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. Technical report, 2007.

[Ioffe and Szegedy, 2015] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[Jampour *et al.*, 2017] Mahdi Jampour, Chen Li, Lap-Fai Yu, Kun Zhou, Stephen Lin, and Horst Bischof. Face inpainting based on high-level facial attributes. *CVIU*, 161:29 – 41, 2017.

[Jia *et al.*, 2014] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Gir-shick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv*, 2014.

[Kalayeh *et al.*, 2017] Mahdi M Kalayeh, Boqing Gong, and Mubarak Shah. Improving facial attribute prediction using semantic segmentation. In *CVPR*, June 2017.

[Kumar *et al.*, 2008] Neeraj Kumar, Peter Belhumeur, and Shree Nayar. Facetracer: A search engine for large collections of images with faces. In *ECCV*. Springer, 2008.

[Kumar *et al.*, 2009] Neeraj Kumar, Alexander C Berg, Peter N Belhumeur, and Shree K Nayar. Attribute and simile classifiers for face verification. In *ICCV*, pages 365–372. IEEE, 2009.

[Le *et al.*, 2012] Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir Bourdev, and Thomas S Huang. Interactive facial feature localization. In *ECCV*. 2012.

[Li *et al.*, 2016] Dangwei Li, Zhang Zhang, Xiaotang Chen, Haibin Ling, and Kaiqi Huang. A richly annotated dataset for pedestrian attribute recognition. *arXiv:1603.07054*, 2016.

[Liu *et al.*, 2015a] Sifei Liu, Jimei Yang, Chang Huang, and Ming-Hsuan Yang. Multi-objective convolutional learning for face labeling. In *CVPR*, June 2015.

[Liu *et al.*, 2015b] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

[Lu *et al.*, 2017] Yongxi Lu, Abhishek Kumar, Shuangfei Zhai, Yu Cheng, Tara Javidi, and Rogerio Feris. Fully-adaptive feature sharing in multi-task networks with applications in person attribute classification. In *CVPR*, 2017.

[Luo *et al.*, 2012] Ping Luo, Xiaogang Wang, and Xiaoou Tang. Hierarchical face parsing via deep learning. In *CVPR*. IEEE, 2012.

[Mo *et al.*,] Zhenyao Mo, John P Lewis, and Ulrich Neumann. Face inpainting with local linear representations.

[Pathak *et al.*, 2016] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

[Rudd *et al.*, 2016] Ethan M Rudd, Manuel Günther, and Terrance E Boult. Moon: A mixed objective optimization network for the recognition of facial attributes. In *ECCV*. Springer, 2016.

[Siddiquie *et al.*, 2011] Behjat Siddiquie, Rogerio S Feris, and Larry S Davis. Image ranking and retrieval based on multi-attribute queries. In *CVPR*. IEEE, 2011.

[Smith *et al.*, 2013] Brandon M Smith, Li Zhang, Jonathan Brandt, Zhe Lin, and Jianchao Yang. Exemplar-based face parsing. In *CVPR*, 2013.

[Wang *et al.*, 2016] Jing Wang, Yu Cheng, and Rogerio Schmidt Feris. Walk and learn: Facial attribute representation learning from egocentric video and contextual data. In *CVPR*, 2016.

[Wang *et al.*, 2017a] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. *arXiv:1711.11585*, 2017.

[Wang *et al.*, 2017b] Zhanxiong Wang, Keke He, Yanwei Fu, Rui Feng, Yu-Gang Jiang, and Xiangyang Xue. Multi-task deep neural network for joint face recognition and facial attribute prediction. In *ICMR*. ACM, 2017.

[Yeh *et al.*, 2017] Raymond A Yeh, Chen Chen, Teck Yian Lim, Alexander G Schwing, Mark Hasegawa-Johnson, and Minh N Do. Semantic image inpainting with deep generative models. In *CVPR*, 2017.

[Zhang *et al.*, 2014] Ning Zhang, Manohar Paluri, Marc'Aurelio Ranzato, Trevor Darrell, and Lubomir Bourdev. Panda: Pose aligned networks for deep attribute modeling. In *CVPR*, 2014.

[Zhong *et al.*, 2016] Yang Zhong, Josephine Sullivan, and Haibo Li. Face attribute prediction using off-the-shelf cnn features. In *ICB*. IEEE, 2016.