Position-Squeeze and Excitation Module for Facial Attribute Analysis

Yan Zhang
452642781@qq.com
Wanxia Shen
51151214005@ecnu.cn
✉Li Sun
sunli@ee.ecnu.edu.cn
Qingli Li
qlli@cs.ecnu.edu.cn

Shanghai Key Laboratory of
Multidimensional Information
Processing,
East China Normal University,
200241 Shanghai, China

Abstract

In this paper, we focus on multiple facial attribute recognition in a single Convolutional Neural Network (CNN). We propose a Position-Squeeze and Excitation (PSE) module, which incorporates the spatial information of different attributes into CNN training. By adding a lateral branch which computes a weight mask for each attribute, the PSE module can help the network learn features from where attributes naturally appear. Moreover, the module can be added as a branch to any classical convolutional neural network to perform end-to-end multi-attribute classification. Experiments show that, our solution has achieved high accuracy on both the CelebA dataset and the LFWA dataset.

1 Introduction

Multi-task learning (MTL) is an interesting but challenging topic. It has been investigated across all applications of machine learning, from natural language processing, speech recognition, computer vision to even drug discovery. In its application in computer vision, predicting multiple facial attributes simultaneously in a supervised way, particularly in CNN, has drawn researchers' attention and achieved high accuracy. The reason of the success lies in following aspects. Firstly, datasets like CelebA and LFWA has been labeled accurately in an intensive way. Multiple attributes of the single face in image are manually labeled. Secondly, these attributes supplement each other's description. So learning them at the same time reduce the over-fitting risk. In addition, CNN has the powerful ability to fit the data in a non-linear way. Successful multiple facial attribute prediction may has a plenty of novel applications, such as descriptive search (*e.g.* searching for white women with blond hair [9, 14]), verification systems, face sorting [13], social sentiment analysis [17] and automatic makeup. Therefore, multiple facial attribute recognition has attracted an increasing number of research [5, 8].

Nevertheless, there are some inherent difficulty in this task. Attributes on people's face are highly correlated and heterogeneous. The correlation between attributes can either be in a positive or negative way. For example, a person who wears makeup or lipstick is more likely to be female and is not likely to have a goatee and beard. In terms of label type, scale, and semantic meanings, individual attributes can be heterogeneous. *E.g*. the attributes of age and hair length are ordered, while gender and race are nominal. These two types of attributes are heterogeneous in terms of label type and scale. Similarly, age, gender and race are reflected by the entire human face, and attributes such as the pointed nose and large lips are the characteristics reflected by the local part of face. These two types of attributes are heterogeneous in terms of semantics. Therefore, the relevance and heterogeneity of attributes should be taken into account, when we are designing a face attribute recognition model. To solve the correlation and heterogeneity problem, Han *et al*. [5] proposed the Deep Multi-task Learning (DMTL) network, making joint estimation by dividing attributes into several groups on the grounds of their relevance and heterogeneity. Attributes share the similar data type, scale or local region in the same group, and it has achieved quite good relsults.

The groups in DMTL is specified based on the prior knowledge, not through data-driven approach. Particularly, when estimating the local attributes, features extracted from the whole face region are used, instead of the specific local region where the specific attribute is located. To identify a particular attribute, humans tend to focus on the position the attribute naturally appears at, although the entire face may also give some cue for that attribute. For instance, to predict whether a person wears a lipstick or not, we should emphasize the mouth region, though the gender reflected on the whole face may provide the cue.

In order to fully consider the local region for some attributes, we propose a Position-Squeeze and Excitation (PSE) module, which helps the network learn the dependency between the attribute and their corresponding local region. Inspired by the SE module [7], which learns a single weight for each channel of the feature maps, the proposed PSE module focuses on highlighting the relevant spatial position for different local attributes on the feature maps. Specifically, the PSE module is incorporated by adding a lateral branch in which the dimension of input feature map is first reduced through Position Average Pooling (PAP) layer, then given to several learn-able layers to generate the weight value for each spatial position in the input feature map. The final weight from the lateral branch excites the input of main branch by multiplication. With PSE module, loss derivative will not be equally propagated to the entire image, but local region will be emphasized or even neglected depending on the weight value calculated on different spatial position. In other words, the PSE module can learn where and how to aggregate and excite the feature maps.

To compare the SE and PSE module, we provide their basic structure in Figure 1. As for PSE module, the first step is *Position-Squeeze*, also named as *Position Average Pooling*. It takes the spatial features $\mathbf{X}$, with its dimension of $C \times W \times H$, to compute the average for each position $F_{sq}(.)$, which is $1 \times H \times W$. The second step is the *Excitation* operation, which leads the network to learn about the location dependecy of the features $F_{ex}(\cdot, W)$, and adjust the feature map $F_{scale}()$ according to the dependency, the adjusted feature map is the output of the PSE module.

In summary, the main contribution of this paper lies in following three aspects. Firstly, we consider the detection of attributes should be based on the location of its natural appearance rather than the entire image domain, we propose the PSE module to assist the network to learn the dependence between features and their locations. Secondly, we applied the PSE module to complete the task of multiple facial feature recognition. Finally, according to the relevance and heterogeneity of facial attributes, we divide multi-attribute recognition task into several subtasks and build a deep multi-task learning network to make joint estimation of multiple attributes in an end-to-end way.

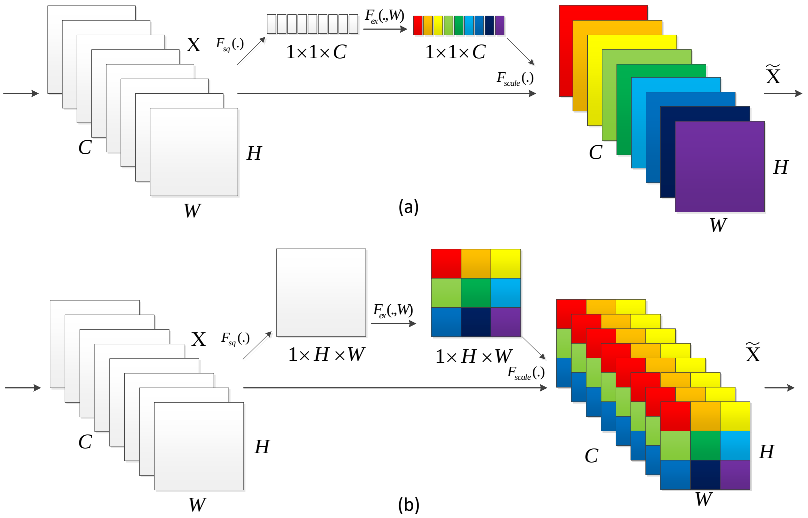

Figure 1: Basic structure of SE module (a) and PSE module (b). SE module focuses on features in different channels, while PSE module pays attention to compute the weight mask for different position.

2 Related Work

In recent years, many research that combine local information and attribute prediction have been carried out. Kalayeh *et al.* [8] proposes a semantic segmentation network to learn local information. Also, they integrate the local information with another attribute analysis network to estimate the facial attributes. The two networks are trained separately, not in an end-to-end way. To be able to use location information to resolve attribute predictions and achieve end-to-end training simultaneously, Ding *et al.* [3] proposes the *hint-based* method, compressing the facial attribute localization network together with the attribute classification network as PaW (Part and Whole) classification network. This method greatly increases the network's complexity and calculation expense. To handle these problems, we propose PSE module. The PSE module can be directly added to the attribute recognition network as a lateral branch so that learning local informaiton and attribute recognition can be done at the same time in an end-to-end way, but will only add a little extra computation cost and network complexity.

3 Our Approach

3.1 Framework Overview

In this paper, our aim is to estimate multiple human face attributes simultaneously through a single multi-head model based on MTL technique. Although recognizing multiple facial attributes at the same time is a big challenge due to the heterogeneity of different attributes, different attributes also share the similar feature representation. MTL network can provide the correlation among attibutes to help make the model robust, and reduce the potential risk of over-fitting. Facial attributes are categorized into two types based on the region it naturally exists in, which are global and local attributes. Attributes like *attractive* and *heavy make-up* are the descriptions of the whole face. Local attributes like *black hair* and *hat* have local spatial heterogeneity. Considering the heterogeneity among local attributes, the joint estimation model should also learn the local spatial difference for different local attributes.

Like other classification task, we also apply CNN in multiple facial attribute recognition. The idea is to make a special structure in CNN which allows it to focus on the different local region the attribute may appear at. Considering the difference among attributes, the heterogeneous attributes can be divided into several groups, and the attribute classification can be made from the specific features of different groups. Figure 2 illustrates the framework of our deep multi-task learning network. We partition all the attributes into different subtasks based on the overall and local spatial region, intending to handle each subtask by embedding the PSE module in the corresponding sub-network. In Figure 2, Sub-network1 focuses on estimating overall attributes, while the other seven sub-networks are for analysing different local attributes, corresponding to the local regions which are hair, eyes, nose, cheek and ears, mouth, chin and neck. The loss function of the eight subtasks is the cross entropy loss function [2].

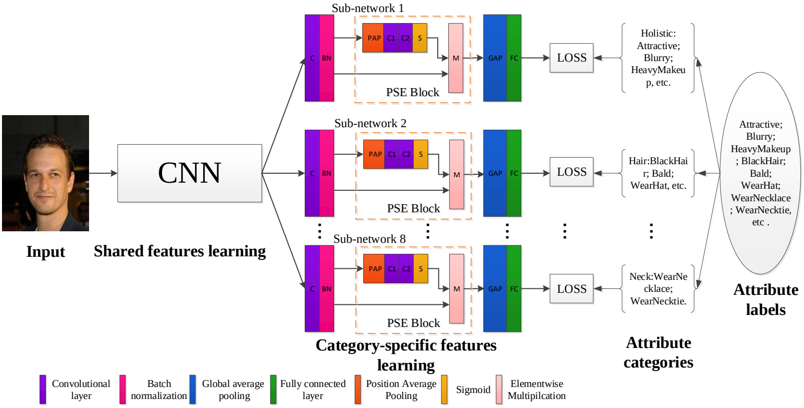
Input
CNN
Shared features learning
Sub-network 1
Sub-network 2
Sub-network 8
C
BN
PAP
C1
C2
S
M
PSE Block
GAP
FC
LOSS
Category-specific features learning
Holistic: Attractive; Blurry; HeavyMakeup, etc.
Hair:BlackHair; Bald; WearHat, etc.
Neck:WearNecklace; WearNecktie.
Attribute categories
Attractive; Blurry; HeavyMakeup; BlackHair; Bald; WearHat; WearNecklace; WearNecktie, etc .
Attribute labels
Convolutional layer
Batch normalization
Global average pooling
Fully connected layer
Position Average Pooling
Sigmoid
Elementwise Multipilcation

Figure 2: DMTL Network consists of several shared layers and 8 sub-networks. Each sub-network has a lateral branch for implementation of PSE module, which computes the weight mask for the corresponding local attribute. The multiplication of the features in the main branch and the output of PSE module enhances the corresponding local information and inhibit the irrelevant one from the whole image domain.

3.2 Position-Squeezed and Excitation Module

PSE module is a computing cell in proposed algorithm, and can be used for any input tensor $\mathbf{X}$, where $\mathbf{X} \in \mathbb{R}^{C \times W \times H}$. Here the tensor has C channels, and with the size of each channel equals $W \times H$. Specifically, we define $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_C]$, and $\mathbf{x}_j = [x_j^1, x_j^2, \ldots, x_j^{W \times H}]$, where the subscript is the channel index and the superscript is the index for spatial position. In order to enhance the expressibility of the features in lateral branch, and inhibit useless one computed from the irrelevant region, the PSE module computes a weight mask for each spatial position in **X**, and the value reflects the relevance of the corresponding position for certain attributes. The PSE module consists of two parts, position squeezing and excitation.

3.2.1 Position Squeezing: embeding spatial information

As features located at the same position in different channels individually, we apply PAP in the lateral branch to compute the average value on each position as the features' location dependency for a certain attribute. Therefore, the spatial information is embedded into the lateral branch. PAP computation is illustrated in Figure 3. Comparing with traditional average pooling within the feature map, PAP intends to encode the spatial information in a single channel feature map, thus producing a spatial distribution map. In formalization, this single channel spatial distribution **Z** is compressed from feature map **X**, $\mathbf{X} \in \mathbb{R}^{C \times W \times H}$. The calculation of **Z** is as follows:

$$\mathbf{Z} = F_{sq}(\sum_j x_j^i) = \frac{1}{C}\sum_{j=1}^{C} x_j^i \qquad i = 1, 2, \cdots, W \times H \tag{1}$$

i represents the pixel index in a single-channel feature map, and j refer to the channel index. $F_{sq}(.)$ is the operation to compute the average of each position in different channels. Here, the simplest averaging method is used to compress feature map into one channel, and of course other classic aggregation strategies can be used. Note that all these computation is derivative and the loss can be backproped through it.

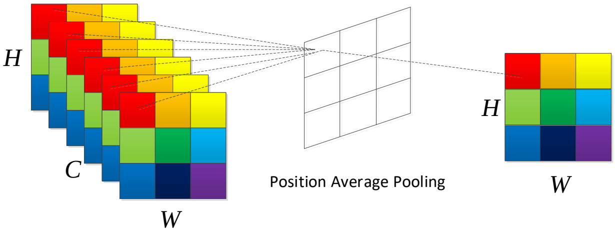

Figure 3: PAP computes the average value of the same position in different channels, and thus we get a single-channel feature map, which demonstrates the spatial distribution.

3.2.2 Excitation: adaptive adjustment

In order to utilize the location information aggregated in the position squeezing operation, the second excitation operation is performed. The purpose of this operation is to fully capture the network's dependency on different locations. The activation function needs to satisfy two conditions: firstly, it must be flexible and can learn the nonlinear interaction correlation between positions. Secondly, it must be able to learn about non-mutually exclusive relationships, as multiple locations may have effects on the outcome. In order to meet these standards, we select the same activation function as the SE module [7], *i.e.* the sigmoid activation function:

$$\mathbf{S} = F_{ex}(\mathbf{Z}, \mathbf{W}) = \sigma(g(\mathbf{Z}, \mathbf{W})) = \sigma(\mathbf{W}_2 \circledast \delta(\mathbf{W}_1 \circledast \mathbf{Z})) \tag{2}$$

$\mathbf{S}$ is a matrix, the dimension of which is $W \times H$, δ represents ReLu function, $\mathbf{W}_1$ and $\mathbf{W}_2$ are variable parameters, and $\circledast$ is the operation of fully connection or convolution. The implementation detail of PSE module is shown in Figure 4.

After the activation of sigmoid function, we get 8 weight masks for their corresponding local region. And then make a spatial elementwise multiplication between the weight mask and the feature map of $\mathbf{X}$ in different channels. We refer to the spatial elementwise multiplication result as $\widetilde{\mathbf{X}}$. The calculation function is as follows:

$$\widetilde{\mathbf{X}} = F_{scale}(\mathbf{X},\mathbf{S}) = \mathbf{S} \odot \mathbf{X} \tag{3}$$

Here, $\widetilde{\mathbf{X}} = [\widetilde{\mathbf{x}}_1, \widetilde{\mathbf{x}}_2, \ldots, \widetilde{\mathbf{x}}_C]$, $\widetilde{\mathbf{x}}_j = [\widetilde{x}_j^1, \widetilde{x}_j^2, \ldots, \widetilde{x}_j^{W\times H}]$, $\mathbf{S} = [s^1, s^2, \ldots, s^{W\times H}]$, F_{scale} is the spatial elementwise multiplication operation, which multiplies the element $\mathbf{x}^i$ at the same position in different channels with the element s^i, located in the corresponding position in $\mathbf{S}$.

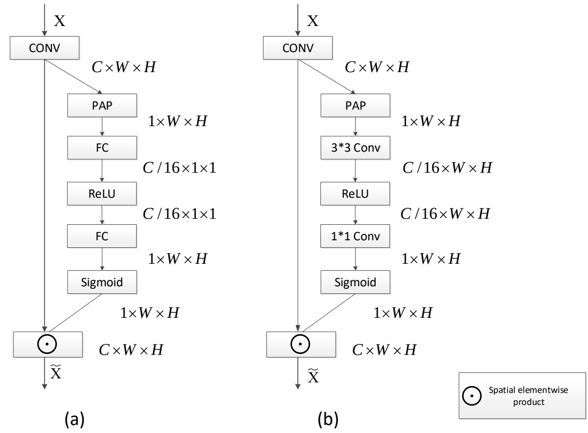

Figure 4: To extract the descriptor of spatial information, we applied the convolution layer and fully connected layer to construct the PSE module respectively. Compared with the FC implementation method, the Convolution method learns less parameters, the evaluation result of the two implementation will be provided in Experiments section.

4 Experiments

4.1 Datasets

We evaluated our method on two challenge face attribute datasets, CelebA [11] and LFWA [16]. CelebFaces Attributes Dataset (CelebA) is a large-scale face attributes dataset with 10177 identities and 202559 face images. It splits 162770 celebrity images for training, 19867 for validation and 19962 for test. Each image has 40 attribute annotations, including *attractive*, *bald*, *arched eyebrow*, *big nose*, etc. It provides *In-The-Wild*, *Align and Cropped* sets, and we applied our method on the aligned one. LFWA dataset contains more than 13,000 images of faces collected from web. It is partitioned into about half for training and half for test. Each image is annotated with exactly the same forty attributes used in CelebA dataset. As LFWA dataset is so small that it will be quite easy to overfit the training set, we add some distortion [1] to the training set, and expand the training set to over 75 thousand images.

4.2 Implementation Details

The 40 attributes of CelebA dataset are all in binary format, and the heterogeneity between these attributes is mainly reflected in the overall and local semantics. We adopt a common CNN architecture attached with 8 sub-networks to get the overall semantic information and 7 different local semantic information respecctively. The division method is the same as that proposed by Han *et al*. [5]. *Attractive*, *Blurry*, *Chubby*, *Heavy Makeup*, *Male*, *Oval Face*, *Pale Skin*, *Smiling* and *Young* are partitioned into overall attributes (sub-network1). Sub-network2 recognizes attributes located at the top of the head. Sub-network3 focuses on eyes. Sub-network4 focuses on nose. Sub-network5 focuses on region around cheek and ears. Sub-network6 is responsible for recognizing attributes in mouth region. And sub-network7 is for chin, while sub-network8 is for neck.

The DMTL framework proposed in this paper, has no restriction on the structure of CNN. We have applied the Alexnet-cvgj [15] and SEnet [7] to build our bottom network, attached PSE module to every branch, and have conducted comparative experiments on the two implementation method of PSE module to evaluate their performance. We mark them as PSE(a)-Alex, PSE(a)-SE, PSE(b)-Alex, and PSE(b)-SE. In addition, we also provide the experiment result of DMTL network without PSE module, referred as Baseline-Alex and Baseline-SE. To assist training, we use the publically available corresponding pre-train model to initialize our net. For all the training images, we first standardize them to $256 \times 256 \times 3$ size, and then randomly left-or-right flip the images in an online way, before they are fed into the network. As for Alexnet-cvgj, we set the initial learning rate to be 0.005, and it will follow a polynomial decay function with the training process going on. Batch size is 256, the max iteration step is 20000, the other hyperparameters are all set to be the same as those of original Alexnet-cvgj model. When training with SEnet, we set the initial learning rate as 0.0001, batch size as 256, the max iteration to be 10000, and the selected optimizer is SGD with momentum equals 0.9. Every 30 epoches, the learning rate will decay by 0.1. The remaining hyperparameters are set the same as the original one. When training on the LFWA dataset, we set the initial learning rate as 0.005, and it will decay by 0.1 every 10000 iteration.

4.3 Results and Analysis

Based on the PSE module and the DMTL framework described above, our results on CelebA dataset is listed in Table 1, and LFWA is listed in Table 2. Results from other current methods are listed as well.

4.3.1 Analysis on PSE Module

As is listed in Table 1, the mean accuracy of Baseline-ALex, PSE(a)-Alex, PSE(b)-Alex are 90.40%, 91.10% and 91.13% respecitively. The results based on SEnet are 90.98%, 91.23%, 91.23% respectively. Under the framework of Alexnet-cvgj, the mean accuracy is increased by 0.70% and 0.73%. Under the framework of SEnet, the mean accuracy is increased by 0.25%. These results prove the PSE module really works for multi-attribute recognition.

Table 1: Results on CelebA

	LNets+ANet [10]	MCNN-AUX [6]	DMTL [5]	AFFACT [4]	PaW [3]	HRP [12]	SSP+SSG [8]	Baseline-Alex	PSE(a)-Alex	PSE(b)-Alex	Baseline-SE	PSE(a)-SE	PSE(b)-SE
5 Shadow	91.00	94.51	95.00	94.21	94.64	93.96	-	94.08	94.57	94.59	94.36	94.85	94.45
Arched Eyebrows	79.00	83.42	86.00	82.12	83.01	83.39	-	82.87	83.33	83.74	83.33	83.75	82.93
Attractive	81.00	83.06	85.00	82.83	82.86	82.86	-	80.69	82.18	82.27	82.05	82.95	82.70
Bags Un Eyes	79.00	84.92	85.00	83.75	84.58	85.12	-	84.13	84.95	84.93	85.14	85.35	85.36
Bald	98.00	98.90	99.00	99.06	98.93	98.46	-	98.83	98.90	98.90	98.98	98.99	99.05
Bangs	95.00	96.05	99.00	96.05	95.93	95.72	-	95.53	95.98	96.04	95.57	95.54	95.54
Big Lips	68.00	71.47	96.00	70.88	71.46	67.29	-	69.91	71.98	71.93	71.48	71.90	72.01
Big Nose	78.00	84.53	85.00	83.82	83.63	83.36	-	82.29	83.68	83.72	84.34	84.09	84.56
Black Hair	88.00	89.78	91.00	90.32	89.84	88.88	-	88.74	90.02	89.90	89.44	89.76	89.63
Blonde Hair	95.00	96.01	96.00	96.07	95.85	95.70	-	95.44	96.00	96.16	95.82	95.50	95.89
Blurry	84.00	96.17	96.00	95.50	96.11	95.87	-	96.14	96.31	96.23	96.24	96.29	96.30
Brown Hair	80.00	89.15	88.00	89.16	88.50	88.42	-	88.62	89.01	89.49	88.64	88.66	87.77
Bushy Eyebrows	90.00	92.84	92.00	92.41	92.62	92.41	-	91.12	91.80	91.45	90.79	91.77	91.12
Chubby	91.00	95.67	96.00	94.98	95.46	94.69	-	95.88	96.03	95.93	95.83	95.94	95.72
Double Chin	92.00	96.32	97.00	96.18	96.26	95.68	-	96.36	96.44	96.38	96.37	96.45	96.32
Eyeglasses	99.00	99.63	99.00	99.61	99.59	99.30	-	99.46	99.55	99.55	99.63	99.65	99.68
Goatee	95.00	97.24	99.00	97.31	97.38	96.70	-	97.35	97.36	97.30	96.59	97.15	97.52
Gray Hair	97.00	98.20	98.00	98.28	98.21	97.57	-	98.20	98.25	98.28	98.10	98.23	98.29
Heavy Makeup	90.00	91.55	92.00	91.10	91.53	91.51	-	86.90	90.10	90.40	90.43	90.39	91.22
H. Cheekbones	87.00	87.58	88.00	86.88	87.44	87.54	-	85.21	87.17	87.63	86.88	87.12	87.36
Male	98.00	98.17	98.00	98.26	98.39	98.14	-	97.90	98.14	98.18	98.16	98.21	98.14
Mouth S. O.	92.00	93.74	94.00	92.60	94.05	93.91	-	92.71	93.38	93.27	93.30	93.85	93.47
Mustache	95.00	96.88	97.00	96.89	96.90	96.12	-	96.38	96.97	96.96	96.88	96.85	96.88
Narrow Eyes	81.00	87.23	90.00	87.23	87.56	86.84	-	86.98	87.46	87.49	87.22	87.35	87.31
No bread	95.00	96.05	97.00	95.99	96.22	96.17	-	95.61	96.01	96.20	95.95	96.22	96.22
Oval Face	66.00	75.84	78.00	75.79	75.03	75.40	-	73.05	72.57	73.03	73.27	73.98	74.20
Pale Skin	91.00	97.05	97.00	97.04	97.08	96.90	-	96.64	97.17	97.16	96.09	97.04	96.92
Pointy Nose	72.00	77.47	78.00	74.83	77.35	76.13	-	76.21	77.44	77.15	77.27	77.30	77.34
Reced. Hairline	89.00	93.81	94.00	93.29	93.44	92.55	-	93.28	93.45	93.31	93.39	93.55	93.40
Rosy Cheeks	90.00	95.16	96.00	94.45	95.07	94.59	-	93.70	95.04	95.05	94.53	95.08	95.20
Sideburns	96.00	97.85	98.00	97.83	97.64	96.83	-	97.57	97.68	97.70	96.87	97.87	97.94
Smiliing	92.00	92.73	94.00	91.77	92.73	92.74	-	90.57	90.97	90.92	91.04	91.00	91.08
Straight Hair	73.00	83.58	85.00	84.10	83.52	83.11	-	82.97	84.38	84.14	84.97	84.93	84.85
Wavy Hair	80.00	83.91	87.00	85.65	84.07	83.28	-	83.06	84.54	84.29	85.34	85.35	86.05
Wear. Earrings	82.00	90.43	91.00	90.20	89.93	90.41	-	89.96	90.65	90.59	90.64	90.26	90.48
Wear. Hat	99.00	99.05	99.00	99.02	99.02	98.71	-	98.90	99.06	98.99	98.95	99.00	99.01
Wear. Lipstick	93.00	94.11	93.00	91.69	94.24	93.23	-	91.89	93.74	93.77	93.18	93.41	94.24
Wear. Necklace	71.00	86.63	89.00	87.85	87.70	87.54	-	86.86	87.00	87.13	87.95	88.01	87.71
Wear. Nectile	93.00	96.51	97.00	96.90	96.85	96.66	-	96.68	96.91	96.94	96.71	97.03	96.92
Young	87.00	88.48	90.00	88.66	88.59	88.45	-	87.30	87.90	88.13	87.49	88.67	88.52
Average	87.30	91.29	**92.60**	91.01	91.23	90.80	81.45	90.40	**91.10**	**91.13**	90.98	**91.23**	**91.23**

Table 2: Results on LFWA

Approach	SSP+SSG [8]	LNets+ANet [10]	Baseline-ALex	PSE(b)-Alex
Average Accuracy	**85.28**	84	84.10	84.45

4.3.2 Analysis on Two Implementation of PSE Module

Although accuracy on PSE (a) and PSE (b) is almost the same, we can still think highly of PSE (b) module. To some extent, fully connection layer will destory the location information of features, but convolution layer can preserve the location context, and will introduce less parameters. So we believe the PSE (b) structure is more consistent with our intuitive idea.

4.3.3 Comparison with Other Approaches

CelebA dataset has two version. Our method and SSP+SSG [8] method are all implemented on the aligned version, while LANets+ANet [10], MCNN-AUX [6], DMTL [5], AFFACT [4], PaW [3] and HRP [12] uses the original version. Clearly, with the same dataset version, the accuracy of our approach is 9.78% higher than that of SSP+SSG method. The rest of the methods use the original version, but they apply better face alignment method to pre-crop the original images. The alignment applied on the aligned version is just a coarse alignment based on eye detection. However, we can see from Table 2, our approach's performance is better than half of them.

4.4 Visulization

PSE module is designed to help the learning of location dependency of features, aiming to extract more useful information from local regions and inhibit useless information in other irrelevant regions. To intuitively show the effect of PSE module, we visualize the output of each branch's PSE module in PSE(b)-Alex framework. The input image is also listed for analysis, as is shown in Figure 5.

According to the position attributes located, we divide the 40 attributes into 8 sub-networks. From sub-network1 to sub-network8, they stand for the whole face, hair, eyes, nose, cheek and ear, mouth, chin, and neck respectively. Compare the heatmap and input image in Figure 5, we can find that the output feature of PSE module in sub-network1, responsible for estimating the overall attribute, has a strong reaction to the whole face region. While the other sub-networks' output have strong reaction to the region the corresponding attributes located at. This quite satisfies our expectation, and it proves PSE module can help network to learn multi-attribute.

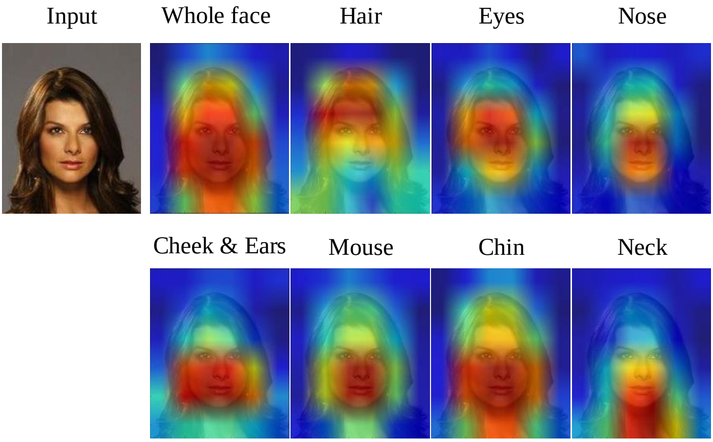

Figure 5: Heatmaps of the corresponding ouputs of 8 PSE module, red represents the activated area, blue represents the inhibited area.

5 Conclusion

In order to focus on the local reigion where attributes naturally appear rather than the whole image domain, we put forward PSE module. PSE module can learn the different location dependency of features, and thus assist each sub-task network to recognize the corresponding attributes. In this paper, we divide the 40 attributes into 8 groups, based on where they can be recognized. We build a DMTL network with 8 sub-networks based on some classical CNN models to make joint multi-attribute recognition. Finally, the experimental results and the visual analysis are presented. It is clearly that the PSE module proposed in this paper is effective and meets the expectations.

6 Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Project 61302125, in part by Natural Science Foundation of Shanghai under Project 17ZR1408500, and in part by the Science and Technology Commission of Shanghai Municipality under Grant No.18511102500. Corresponding to sunli@ee.ecnu.edu.cn.

References

[1] M. D. Bloice, C. Stocker, and A. Holzinger. Augmentor: An Image Augmentation Library for Machine Learning. *ArXiv e-prints*, August 2017.

[2] Pieter Tjerk De Boer, Dirk P. Kroese, Shie Mannor, and Reuven Y. Rubinstein. A tutorial on the cross-entropy method. *Annals of Operations Research*, 134(1):19–67, 2005.

[3] Hui Ding, Hao Zhou, Shaohua Kevin Zhou, and Rama Chellappa. A deep cascade network for unaligned face attribute classification. 2017.

[4] Manuel GÃijnther, Andras Rozsa, and Terrance E. Boult. Affact - alignment free facial attribute classification technique. 2016.

[5] H. Han, K Jain A, S. Shan, and X. Chen. Heterogeneous face attribute estimation: A deep multi-task learning approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–1, 2017.

[6] Emily M Hand and Rama Chellappa. Attributes for improved attributes: A multi-task network for attribute classification. 2016.

[7] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. 2017.

[8] Mahdi M. Kalayeh, Boqing Gong, and Mubarak Shah. Improving facial attribute prediction using semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4227–4235, 2017.

[9] Neeraj Kumar, Alexander Berg, Peter N. Belhumeur, and Shree Nayar. Describable visual attributes for face verification and image search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):1962–1977, 2011.

[10] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. pages 3730–3738, 2014.

[11] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[12] U. Mahbub, S. Sarkar, and R. Chellappa. Segment-based Methods for Facial Attribute Detection from Partial Faces. *ArXiv e-prints*, January 2018.

[13] Devi Parikh and Kristen Grauman. Interactively building a discriminative vocabulary of nameable attributes. In *Computer Vision and Pattern Recognition*, pages 1681–1688, 2011.

[14] W. J. Scheirer, Neeraj Kumar, P. N. Belhumeur, and Terrance E. Boult. Multi-attribute spaces: Calibration for attribute fusion and similarity search. 157(10):2933–2940, 2012.

[15] Marcel Simon, Erik Rodner, and Joachim Denzler. Imagenet pre-trained models with batch normalization. 2016.

[16] L Wolf, T Hassner, and Y Taigman. Effective unconstrained face recognition by combining multiple descriptors and learned background statistics. *IEEE Trans Pattern Anal Mach Intell*, 33(10):1978–1990, 2011.

[17] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Learning social relation traits from face images. In *IEEE International Conference on Computer Vision*, pages 3631–3639, 2015.