Relation-Aware Pedestrian Attribute Recognition with Graph Convolutional Networks

Zichang Tan,[1,2] Yang Yang,[1,2] Jun Wan,[1,2*] Guodong Guo,[3,4] Stan Z. Li[1,2,5]
[1]CBSR&NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2]University of Chinese Academy of Sciences, Beijing, China
[3]Institute of Deep Learning, Baidu Research, Beijing, China
[4]National Engineering Laboratory for Deep Learning Technology and Application, Beijing, China
[5]Faculty of Information Technology, Macau University of Science and Technology, Macau, China
{zichang.tan, yang.yang, jun.wan, szli}@nlpr.ia.ac.cn, guoguodong01@baidu.com

Abstract

In this paper, we propose a new end-to-end network, named Joint Learning of Attribute and Contextual relations (JLAC), to solve the task of pedestrian attribute recognition. It includes two novel modules: Attribute Relation Module (ARM) and Contextual Relation Module (CRM). For ARM, we construct an attribute graph with attribute-specific features which are learned by the constrained losses, and further use Graph Convolutional Network (GCN) to explore the correlations among multiple attributes. For CRM, we first propose a graph projection scheme to project the 2-D feature map into a set of nodes from different image regions, and then employ GCN to explore the contextual relations among those regions. Since the relation information in the above two modules is correlated and complementary, we incorporate them into a unified framework to learn both together. Experiments on three benchmarks, including PA-100K, RAP, PETA attribute datasets, demonstrate the effectiveness of the proposed JLAC.

Introduction

Visual analysis of pedestrian attributes, *e.g.*, gender, age and body shape, has received increasing attention in recent years (Wang et al. 2017; Sarfraz et al. 2017; Lin et al. 2019; Liu et al. 2017; Zhao et al. 2018; Xiang et al. 2019), due to its wide range of potential applications, such as person retrieval (Feris et al. 2014), person re-identification (Lin et al. 2019; Zheng et al. 2015) and so on. Although significant efforts have been devoted to pedestrian attribute recognition, it remains a challenging problem because of low resolution, occlusions and complex variations (e.g., human poses, camera viewing angles and background) in surveillance scenes.

In the field of pedestrian attribute recognition, dozens of attributes are often required to be analyzed together, like gender, age, sunglasses, clothing types and hair style. In those attributes, some of them are closely related. For example, the attribute "skirt" is often associated with the attribute "female", and the attribute of clothing types can provide certain information for judging the age. Observed with this phenomenon, a reliable solution of enhancing the recognition performance is to explore the relations among multiple pedestrian attributes. Most previous works (Wang et al. 2017; Sarfraz et al. 2017; Lin et al. 2019; Liu et al. 2018; Li et al. 2018a) exploit the relations among multiple attributes by only using a simple multi-task learning (MTL) framework, where the information exchanges among different attributes are only allowed in the shared low-level layers. Since MTL employs the losses followed by the final layers to guide its learning, the explicit information exchanges and propagation among different attributes may be insufficient. Thus, such a framework lacks a thorough and comprehensive representation of the relations among attributes.

Exploration of the contextual relations in different image regions is also helpful for attribute recognition. A conceivable example is that when recognizing the gender of a pedestrian, one tends to focus on multiple regions like the regions around the head, human body and carrying things, and consider their contextual relations. Although the deep convolutional networks have achieved a great success in pedestrian attribute recognition, the contextual relations have not been fully exploited. This is because that the receptive fields of units in those deep convolutional networks are severely limited according to the work (Luo et al. 2016), which may fail to learn the global context and capture the long range dependencies in different regions.

To deal with the above-mentioned problems, we resort to Graph Convolutional Network (GCN) (Kipf and Welling 2017) which has a strong ability of modeling the dependencies and propagating messages between the concepts on a graph structure. Specifically, we construct two graph modules, named attribute relation module (ARM) and contextual relation module (CRM), to discover and capture the attribute and contextual relations, respectively. In these two modules, the key issue is how to construct the graph structure. In ARM, we first learn the attribute-specific features by the constrained losses with each feature corresponding to an attribute. Then, each learned feature would be treated as a node in the graph. In CRM, we define the clusters of regions/pixels as the nodes of the graph. In consideration of the variations of human poses and camera viewing angles, we let the network learn to cluster the regions by itself rather than using the predefined regions as in the previous

*Corresponding Author

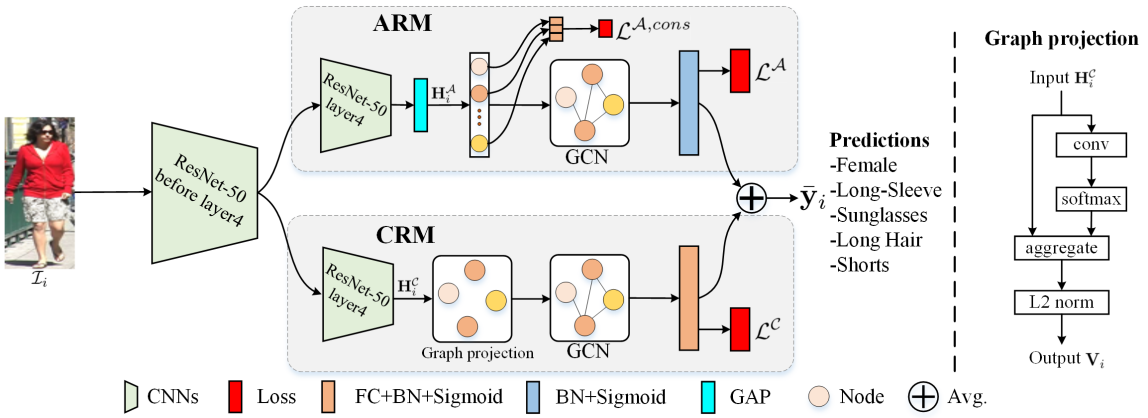

Figure 1: An overview of the proposed JLAC.

work (Li et al. 2019b). In those two graphs, messages are allowed to propagate among their nodes and the GCN layer is further employed to capture the relations among them. To synergize the above two modules together, we further formulate a two-branch network by incorporating them in parallel, where both of them are learned jointly and concurrently to fully capture those relations.

The main contributions of our work are as follows: (1) We propose a novel end-to-end unified framework, which employs GCN to capture both the attribute and contextual relations for pedestrian attribute recognition; (2) We present two novel ways of constructing the graph structures, where one is to formulate the graph by using the attribute-specific features learned by the constrained losses, and the other defines the clusters of regions/pixels as the nodes by using a graph projection scheme. To the best of our knowledge, such two approaches for graph construction the graph have not been studied in the literature of pedestrian attribute recognition; (3) We achieve the new state-of-the-art performance on three benchmark datasets of pedestrian attribute recognition, including RAP, PETA and PA-100K datasets.

Related Works

Pedestrian Attribute Recognition: Recently, deep learning (Simonyan and Zisserman 2015; He et al. 2016; Tan et al. 2019a) has achieved great successes in pedestrian attribute recognition (Wang et al. 2017; Sarfraz et al. 2017; Liu et al. 2018; Zhao et al. 2019; Li et al. 2019b; Tan et al. 2019b; Li et al. 2019a). Previous works mainly solve the task of pedestrian attribute recognition by formulating attention mechanisms (Liu et al. 2017; Sarafianos, Xu, and Kakadiaris 2018; Zhao et al. 2019; Tan et al. 2019b), utilizing the pose or body parts information (Liu et al. 2018; Li et al. 2018a; Zhao et al. 2018), or coping with the imbalance data problem (Sarafianos, Xu, and Kakadiaris 2018; Wang et al. 2019). Most previous models (Sarfraz et al. 2017; Lin et al. 2019; Liu et al. 2018; Li et al. 2018a) are constructed based on the MTL framework, where the relation exploration among multiple attributes is still not sufficient due to the lack of an explicit information propagation mechanism among different attributes. Wang et al. (Wang et al. 2017) propose a sequential recurrent model to explore the attribute relations. However, this model depends on the sequential order of attributes and an ensemble of multiple models is employed for improving the performance. In contrast, the graph structure in our work is less dependent on the attribute orders and explores the relations more fully and efficiently. On the other hand, Li et al. (Li et al. 2019b) learn the attribute-attribute relations based on the word embedding of attribute names. However, such learned relations only reflect the literal meaning instead of image's content. Different from their work, we explore the attribute relations based on attribute-specific features learned by the network itself, which contains more abundant information. Further, the contextual relations from different image regions are also important for pedestrian attribute recognition. Zhao et al. (Zhao et al. 2018) employ a Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber 1997) to capture the relations among different body parts. Li et al. (Li et al. 2019b) capture the spatial relations by simply dividing the image into rigid grids. Different with the above works, we propose a graph projection scheme to project a 2-D feature map into a set of nodes and then construct an undirected graph based on them to capture the contextual relations of different image regions.

Neural Networks with Graphs: Graph Neural Networks (GNN), which are proposed in the works (Gori, Monfardini, and Scarselli 2005; Scarselli et al. 2009), are capable of extending the neural networks to process the data with a graph structure. After that, various methods based on GNN are proposed, e.g., Gated Graph Neural Networks (GGNN) (Li et al. 2016), Graph Attention networks (GATs) (Velivckovic et al. 2018) and GCN (Kipf and Welling 2017). Our work may be most related to GCN (Kipf and Welling 2017), which is originally proposed for semi-supervised learning. In recent years, some researchers have focused on GCN for image classification (Chen et al. 2019), human action recognition (Yan, Xiong, and Lin 2018), semantic segmentation (Li and Gupta 2018), pedestrian analysis (Li et al. 2019b; 2019a) and so on. Different from those works, we adop-

t GCN to explore the attribute relations and the contextual relations among different image regions.

Our Approach

We first introduce the notations and give an overview of our approach. Then, we present the preliminaries of GCN, and present the two proposed GCN-based modules, namely ARM and CRM. Finally, the employed loss function is given and explained.

Notations and Overview

In our approach, suppose the training set contains m samples and is denoted as $\mathcal{D} = \{\mathcal{I}_i, \{y_{ij}\}_{j=0}^{\mathcal{S}-1}\}$, $i = 0, \ldots, m-1$, where y_{ij} represents the label for the j^{th} attribute of image $\mathcal{I}_i$, and $\mathcal{S}$ indicates the number of attributes. The pipeline of the proposed architecture is shown in Fig. 1. The proposed JLAC is constructed based on a two-branch network with the backbone of ResNet-50 (He et al. 2016), where the main body of ResNet-50 is shared except for the last three residual units. Those two branches are unshared and each branch employs the last three residual units of ResNet-50 as the main architecture. For the i^{th} image $\mathcal{I}_i$, it first passes through the two-branch ResNet-50 to obtain the high-level features $\mathbf{H}_i^{\mathcal{A}}$ and $\mathbf{H}_i^{\mathcal{C}}$, where the superscripts $\mathcal{A}$ and $\mathcal{C}$ refer to ARM and CRM, respectively. Then, two GCN-based modules, namely ARM and CRM, are designed to exploit the attribute and contextual relations. Finally, the predictions from those two modules are fused together to get the final predictions.

A Brief Introduction to GCN

GCN (Kipf and Welling 2017) propagates the messages on the graph structure and can efficiently learn the relations among graph nodes. Suppose the graph has n nodes. It takes the node features $\mathbf{Z} \in R^{n \times d}$, and the corresponding adjacency matrix $\mathbf{A} \in R^{n \times n}$ as inputs, where d denotes the dimension of input features. Mathematically, a linear graph convolution is represented as:

$$\tilde{\mathbf{Z}} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I})\mathbf{D}^{-\frac{1}{2}}\mathbf{Z}\mathbf{W} \tag{1}$$

In the above equation, $\mathbf{D}_{ii} = \sum_j (\mathbf{A}_{ij} + \mathbf{I}_{ij})$ where $\mathbf{I}$ is an identity matrix, and $\mathbf{W}$ is the filter matrix of the graph convolution layer to be learned. $\mathbf{A} + \mathbf{I}$ denotes the adjacent matrix with self-connections, and the left part $\mathbf{D}^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I})\mathbf{D}^{-\frac{1}{2}}$ is used to normalize the adjacent matrix (Kipf and Welling 2017). The non-linear activations like ReLU or sigmoid can be appended to the convolutional layer to capture more effective features. For more details, readers can refer to the work (Kipf and Welling 2017).

Attribute Relation Module

This module is designed to discover and capture the attribute relations. It first extracts the attribute-specific features with $\mathcal{S}$ fully connected layers based on the high-level features $\mathbf{H}_i^{\mathcal{A}} \in R^c$ (obtained after Global Average Pooling (GAP) layer, and $c = 2048$ denotes the number of feature channels), where each fully connected layer corresponds to a specific attribute. For convenience, the attribute-specific features of the j^{th} attribute for the i^{th} image are denoted as $\mathbf{x}_{ij} \in R^d$ with d as its dimensionality. In our implementation, the predicted score of the j^{th} attribute is only generated from $\mathbf{x}_{ij}$, which ensures its learning is only under the label supervision of the j^{th} attribute. The predicted score is mathematically represented as:

$$\hat{y}_{ij}^{\mathcal{A},cons} = \sigma(BN((\mathbf{w}_j^{\mathcal{A},cons})^T \mathbf{x}_{ij})) \tag{2}$$

where $\mathbf{w}_j^{\mathcal{A},cons}$ denotes the parameters of the j^{th} attribute in the classifier, σ is a sigmoid function and BN represents a batch normalization (BN) (Ioffe and Szegedy 2015) layer. The BN layer is used to balance the positive and negative outputs according to the works (Zhao et al. 2018; 2019), which can alleviate the imbalanced data problem. It first normalizes the predicted vectors with a zero mean and a unit variance, and then learns to scale its value and add an appropriate bias to it. Thus, it changes the output distributions of the positive and negative samples, and makes the output distribution be adapted to the unbalanced data for achieving better performance. The predicted scores are then used to compute the constrained losses which will be introduced in the Section of *The Loss Function*.

Given the attribute-specific features $\mathbf{X}_i \in R^{\mathcal{S} \times d}$ which is a matrix form of $\{\mathbf{x}_{ij}\}_{j=0}^{\mathcal{S}-1}$, we construct a GCN layer by taking it as the input to explore the relations among multiple pedestrian attributes, which can be implemented according to the following formula:

$$\tilde{\mathbf{X}}_i = \mathbf{D}^{\mathcal{A}^{-\frac{1}{2}}}(\mathbf{A}^{\mathcal{A}} + \mathbf{I})\mathbf{D}^{\mathcal{A}^{-\frac{1}{2}}}\mathbf{X}_i\mathbf{W}^{\mathcal{A}} \tag{3}$$

where $\mathbf{A}^{\mathcal{A}}$ is a learnable adjacency matrix (Yan, Xiong, and Lin 2018). The graph convolution is implemented by performing a convolution with the filter $\mathbf{W}^{\mathcal{A}}$ and then multiplying the input features with the normalized adjacent matrix $\mathbf{D}^{\mathcal{A}^{-\frac{1}{2}}}(\mathbf{A}^{\mathcal{A}} + \mathbf{I})\mathbf{D}^{\mathcal{A}^{-\frac{1}{2}}}$. The updated nodes are generated by utilizing the messages from all nodes. In our implementation, we set $\mathbf{W}^{\mathcal{A}}$ as the size of $d \times 1$, and the output of this GCN layer is an $\mathcal{S} \times 1$ matrix with each output corresponding to an attribute. The predicted scores can be generated by using a BN layer and a sigmoid activation based on $\tilde{\mathbf{X}}_i$, which is formulated as:

$$\hat{\mathbf{y}}_i^{\mathcal{A}} = \sigma(BN(\tilde{\mathbf{X}}_i)) \tag{4}$$

Contextual Relation Module

This module aims to explore the contextual relations among different image regions. Given the input features $\mathbf{H}_i^{\mathcal{C}} \in R^{c \times hw}$ (obtained before GAP layer, and h, w denote the height and width of the feature map, respectively), we first employ a graph projection scheme to project the 2-D feature map to a set of clusters/nodes, with the number of v prespecified. Inspired by NetVLAD (Arandjelovic et al. 2016), we first adopt a soft-assignment scheme to calculate the weight for assigning the p^{th} pixel $\mathbf{H}_{ip}^{\mathcal{C}}$ to the k^{th} cluster, which is written as:

$$\alpha_{ip}^k = \frac{\exp((\mathbf{w}_k^{\mathcal{C}})^T \mathbf{H}_{ip}^{\mathcal{C}} + b_k)}{\sum_l \exp((\mathbf{w}_l^{\mathcal{C}})^T \mathbf{H}_{ip}^{\mathcal{C}} + b_l)} \tag{5}$$

where it is achieved by a softmax function, and $\mathbf{w}_k^{\mathcal{C}}$ and b_k are the trainable parameters for the k^{th} cluster. After that, given v learnable anchor points $\{\mathbf{c}_k\}_{k=0}^{v-1}$, we aggregate the features for the k^{th} node by using the weighted average of the residuals between input features $\mathbf{H}_i^{\mathcal{C}}$ and the anchor point $\mathbf{c}_k$, which can be represented as:

$$\tilde{\mathbf{v}}_{ik} = \frac{1}{\sum_p \alpha_{ip}^k} \sum\nolimits_p \alpha_{ip}^k (\mathbf{H}_{ip}^{\mathcal{C}} - \mathbf{c}_k) \tag{6}$$

Then, the aggregated features $\tilde{\mathbf{v}}_{ik} \in R^c$ are further normalized by L2 normalization, which can be written as $\mathbf{v}_{ik} = \tilde{\mathbf{v}}_{ik} / \|\tilde{\mathbf{v}}_{ik}\|_2$. In this way, different nodes contain features from different image regions. The features of all vertices $\{\mathbf{v}_{ik}\}_{k=0}^{v-1}$ also can be represented as a matrix form $\mathbf{V}_i \in R^{v \times c}$ with each row representing a graph node.

To capture the contextual relations among different regions, we consider a graph with v nodes based on the features $\mathbf{V}_i$. A GCN layer is employed to propagate the messages among all nodes and update their states according to the following formula:

$$\tilde{\mathbf{V}}_i = \varphi(\mathbf{D}^{\mathcal{C}^{-\frac{1}{2}}}(\mathbf{A}^{\mathcal{C}} + \mathbf{I})\mathbf{D}^{\mathcal{C}^{-\frac{1}{2}}}\mathbf{V}_i\mathbf{W}^{\mathcal{C}}) \tag{7}$$

where φ denotes the ReLU function, and $\mathbf{A}^{\mathcal{C}}$ is a learnable adjacent matrix according to the work (Yan, Xiong, and Lin 2018). Then, we concatenate the updated states of all nodes $\{\tilde{\mathbf{v}}_{ik}\}_{k=0}^{v-1}$ and denote it as $\bar{\mathbf{v}}_i$. The predicted scores are obtained based on the updated states as following:

$$\hat{\mathbf{y}}_i^{\mathcal{C}} = \sigma(BN((\mathbf{w}^{\mathcal{C}})^T \bar{\mathbf{v}}_i)) \tag{8}$$

where $\mathbf{w}^{\mathcal{C}}$ indicates the parameters of the classifiers.

The Loss Function

The losses employed to guide the whole network training are three folds. One is the constrained losses of learning the attribute-specific features $\{\mathbf{x}_{ij}\}_{j=0}^{\mathcal{S}-1}$, and the other two are the losses of training ARM and CRM modules. In our experiments, all classifiers employ the binary cross-entropy loss as the loss function. We take the constrained loss for the j^{th} attribute as an example, which can be written as:

$$\begin{aligned} \mathcal{L}_j^{\mathcal{A},cons} = -\frac{1}{m} \sum_{i=0}^{m-1} \rho_{ij} \Big(& y_{ij} \log(\hat{y}_{ij}^{\mathcal{A},cons}) \\ & + (1 - y_{ij}) \log(1 - \hat{y}_{ij}^{\mathcal{A},cons}) \Big) \end{aligned} \tag{9}$$

where ρ_{ij} is a penalty coefficient used to alleviate the imbalanced data problem in pedestrian attribute recognition. Suppose r_j represents the ratio of the images with the j^{th} attribute, and we set $\rho_{ij} = \sqrt{\frac{1}{2r_j}}$, if $y_{ij} = 1$; otherwise $\rho_{ij} = \sqrt{\frac{1}{2(1-r_j)}}$. To be specific, for a positive example, ρ_{ij} becomes larger along with r_j decreasing, which shifts the bias of the classifier to favor the minority class. The sum of losses for all attributes can be denoted as $\mathcal{L}^{\mathcal{A},cons} = \sum_j \mathcal{L}_j^{\mathcal{A},cons}$. The losses for training ARM and CRM modules are produced in a similar way, denoted as $\mathcal{L}^{\mathcal{A}}$ and $\mathcal{L}^{\mathcal{C}}$, respectively. The total loss of training the whole network can be represented as:

$$\mathcal{L} = \lambda_1 \mathcal{L}^{\mathcal{A},cons} + \lambda_2 \mathcal{L}^{\mathcal{A}} + \lambda_3 \mathcal{L}^{\mathcal{C}} \tag{10}$$

where $\lambda_1, \lambda_2, \lambda_3$ are weight parameters for those losses. Considering that the branch of CRM is trained only under the supervision of $\mathcal{L}^{\mathcal{C}}$, we simply set its weight λ_3 to 1. However, the branch of ARM is trained under the supervision of both $\mathcal{L}^{\mathcal{A},cons}$ and $\mathcal{L}^{\mathcal{A}}$, and thus the values of the corresponding loss weights λ_1 and λ_2 are selected by experiments (see experimental part). Those losses are optimized jointly and concurrently to fully explore the attribute and contextual relations for pedestrian attribute recognition. In the inference stage, the average results of $\hat{\mathbf{y}}_i^{\mathcal{A}}$ and $\hat{\mathbf{y}}_i^{\mathcal{C}}$ are employed to generate the final predictions, which is denoted as: $\bar{\mathbf{y}}_i = \frac{\hat{\mathbf{y}}_i^{\mathcal{A}} + \hat{\mathbf{y}}_i^{\mathcal{C}}}{2}$.

Experiments

We first introduce the datasets, settings and evaluation metrics for experiments. Then, we present the experimental results and analysis to validate the effectiveness of JLAC.

Datasets and Metrics

We conduct experiments for pedestrian attribute recognition on three benchmark datasets: PA-100K (Liu et al. 2017), RAP (Li et al. 2018b) and PETA (Deng et al. 2014) datasets. **PA-100K dataset** is the largest pedestrian attribute dataset with 100,000 pedestrian images from various outdoor scenes. It provides the annotations of 26 commonly used attributes. Following the settings in (Liu et al. 2017), the dataset is divided into three subsets with 80,000, 10,000 and 10,000 images for training, validation and test, respectively. **RAP dataset** is the largest pedestrian attribute dataset of indoor scenes, and it contains 41,585 pedestrian images. We follow the official protocol provided by Li et al. (Li et al. 2018b) to only select 51 attributes for evaluation. The model is evaluated with 5 random splits, where 33,268 images are used for training and 8,317 images for test in each split. The averaging performance over all splits is used for final evaluation. Moreover, **PETA dataset** is a widely used dataset for pedestrian attribute reccognition. It collects 19,000 images from various outdoor scenes. According to the work (Deng et al. 2014), 35 binary attributes are selected for evaluation. The dataset is randomly split into 3 parts, where 9,500 images are used for training, 1,900 images for validation and the rest 7,600 images for test.

According to the works (Liu et al. 2017; Li et al. 2018b), we adopt five criteria to evaluate the model on PA-100K, PETA and RAP datasets, including a label-based criterion mean accuracy (mA), and four instance-based criteria accuracy (Accu), precision (Prec), Recall and F1. Those metrics are widely used in pedestrian attribute recognition.

Experimental Settings

In our experiments, we adopt the image with the size of 256×128 as input. Before feeding the images to the network, all images are normalized by subtracting a mean and divide a standard deviation for each color channel. In the training

Table 1: Ablation studies on PETA, RAP and PA-100K datasets.

Model	PETA					RAP					PA-100K				
	mA	Accu	Prec	Recall	F1	mA	Accu	Prec	Recall	F1	mA	Accu	Prec	Recall	F1
PAR	84.81	77.66	85.43	85.85	85.64	80.49	66.00	76.35	81.32	78.76	81.17	77.84	85.95	87.19	86.56
PAR + ARM	85.75	79.15	86.12	87.13	86.62	82.86	67.79	76.86	83.48	80.04	82.31	78.48	85.89	88.12	86.99
PAR + CRM	85.63	79.32	86.80	86.65	86.73	83.20	68.63	78.19	83.18	80.61	81.84	78.31	86.48	87.29	86.89
PAR + ARM + CRM	86.03	80.11	87.50	87.09	87.29	83.21	68.85	78.40	83.27	80.77	82.10	79.13	86.91	88.01	87.46
PAR + ARM + CRM + BN	86.96	80.38	87.81	87.09	87.45	83.69	69.15	79.31	82.40	80.82	82.31	79.47	87.45	87.77	87.61

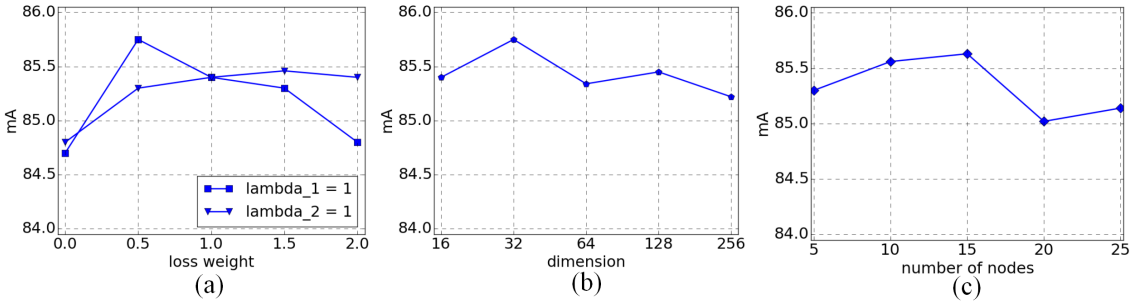

Figure 2: The results of varying the values of (a) loss weights λ_1 and λ_2, (b) the feature dimension d of each node in ARM and (c) the number of nodes v in CRM.

stage, data augmentation is also employed to improve the performance. We augment the training images with horizontal flipping, random scaling, rotation, translation, cropping, erasing and adding random gaussian blurs. Note that only the random horizontal flipping is conducted when evaluating on PA-100K, because it contains a large number of training images. To obtain the features with a large size for CRM, we remove the $\times\frac{1}{2}$ downsampling in the residual uints of CRM. In this way, the output features $\mathbf{H}_i^{\mathcal{C}}$ has the height h and width w of 16 and 8, respectively. All networks are first pretrained on the ImageNet (Deng et al. 2009), and then finetuned on pedestrian attribute datasets. All networks are optimized by Adam optimizer (Kingma and Ba 2015) with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. The learning rate is started with 0.0001 and reduced by a factor of 10 when the number of iterations increases.

Parameter Analysis

In this sub-section, we aim to investigate the effect of varying the values of some parameters in the proposed JLAC, including the loss weights λ_1 and λ_2, the feature dimension d of each node in ARM and the number of nodes v in CRM. The experiments are taken on the PETA dataset and mA is used as the metric for analysis.

Influence of Loss Weights λ_1 and λ_2 represent the trade-off between the losses $\mathcal{L}^{\mathcal{A},cons}$ and $\mathcal{L}^{\mathcal{A}}$. Instead of using a simple grid searching strategy which needs lots of experiments, we first set $\lambda_1 = 1$ and vary the values of λ_2, and then set $\lambda_2 = 1$ and vary the values of λ_1. By doing so, we can reduce a lot of efforts. The experimental results are shown in Fig. 2 (a). We can find the model performs well when using a slightly large λ_1 and a slightly small λ_2, where the well-learned attribute-specific features may facilitate to explore the attribute relations. The model achieves the highest performance when $\lambda_1 = 1$ and $\lambda_2 = 0.5$, which are adopted in other experiments. Moreover, when we only use the constrained loss $\mathcal{L}^{\mathcal{A},cons}$ (with $\lambda_1 = 1$ and $\lambda_2 = 0$) or the loss $\mathcal{L}^{\mathcal{A}}$ (with $\lambda_1 = 0$ and $\lambda_2 = 1$), the performance of the model is poor where the attribute relations are hardly to be captured. This indicates that the attribute relations can be captured only when both two losses are employed together. One is to learn the attribute-specific features while the other is to help the GCN module to learn their relations.

Table 2: The comparisons on PETA dataset.

Method	mA	Accu	Prec	Recall	F1
CNN+SVM	76.65	45.41	51.33	75.14	61.00
ACN	81.15	73.66	84.06	81.26	82.64
DeepMar	82.89	75.07	83.68	83.14	83.41
HP-net	81.77	76.13	84.92	83.24	84.07
VeSPA	83.45	77.73	86.18	84.81	85.49
JRL	85.67	–	86.03	85.34	85.42
Fusion	82.97	78.08	86.86	84.68	85.76
VAA	84.59	78.56	86.79	86.12	86.46
GRL	86.70	–	84.34	**88.82**	86.51
RA	86.11	–	84.69	88.51	86.56
JLPLS-PAA	84.88	79.46	87.42	86.33	86.87
JLAC (ours)	**86.96**	**80.38**	**87.81**	87.09	**87.45**

Analysis on Feature Dimension The feature dimension d of graph nodes in ARM also needs to be studied. The experimental results with different d are shown in Table 2 (b). The model performs best when $d = 32$, and it is used in other experiments. When d is too small, each node may hardly retain sufficient information for later attribute recognition. When d is too large, it contains much redundant information, which is also invalid for later learning.

Table 3: The comparisons on RAP dataset.

Method	mA	Accu	Prec	Recall	F1
CNN+SVM	72.28	31.72	35.75	71.78	47.73
ACN	69.66	62.61	80.12	72.26	75.98
DeepMar	73.79	62.02	74.92	76.21	75.56
VeSPA	77.70	67.35	79.51	79.67	79.59
HP-net	76.12	65.39	77.33	78.79	78.05
JRL	77.81	–	78.11	78.98	78.58
Fusion	74.31	64.57	78.86	75.90	77.35
LG-Net	78.68	68.00	80.36	79.82	80.09
GRL	81.20	–	77.70	80.90	79.29
RA	81.16	–	79.45	79.23	79.34
JLPLS-PAA	81.25	67.91	78.56	81.45	79.98
CoCNN	81.42	68.37	**81.04**	80.27	80.65
JLAC (ours)	**83.69**	**69.15**	79.31	**82.40**	**80.82**

Table 4: The comparisons on PA-100K dataset.

Method	mA	Accu	Prec	Recall	F1
DeepMar	72.70	70.39	82.24	80.42	81.32
HP-net	74.21	72.19	82.97	82.09	82.53
Fusion	74.95	73.08	84.36	82.24	83.29
LG-Net	76.96	75.55	86.99	83.17	85.04
JLPLS-PAA	81.61	78.89	86.83	87.73	87.27
CoCNN	80.56	78.30	**89.49**	84.36	86.85
JLAC (ours)	**82.31**	**79.47**	87.45	**87.77**	**87.61**

Effect of the Number of Nodes We also conduct the experiments by varying the number of nodes v in CRM, and the results are shown in Fig. 2 (c). The conclusion is natural, where both containing too few or too many nodes would result to poor performance, due to the graph has limited capacity or contains much redundant information, respectively. The highest performance is achieved when v is set to 15, and this value is used in other experiments.

Ablation Studies on Components

In this sub-section, we investigate the performance of three employed components, including ARM, CRM and the BN operation in the classifiers. The experiments are conducted on PETA, RAP and PA-100K datasets. The plain MTL framework with a shared representation is employed as the baseline (at this time, none of the relation modules is employed), and we denote it as Pedestrian Attribute Recognition (PAR). The experimental results are shown in Table 1.

For ARM, it improves the performance on all three datasets compared with the baseline, where the mean performance over five criteria is improved by 1.08%, 1.62%, 0.62% on PETA, RAP and PA-100K dataset, respectively. It shows that the attribute relations are more fully exploited compared with the plain MTL network. For CRM, it improves the mean performance over five criteria by 1.15%, 2.18% and 0.42% on PETA, RAP and PA-100K datasets, respectively. It demonstrates that exploring the contextual relations helps to extract more effective features. When both ARM and CRM modules are employed, the mean performance can be further improved on all three datasets, which shows that the complementary and correlated features are learned. Moreover, when BN layer is further adopted, we find that the mean performance is improved by 0.33%, 0.17% and 0.20% on PETA, RAP and PA-100K datasets respectively. The BN layer inserted to all classifiers is to balance the positive and negative outputs, which alleviates the imbalanced data problem in pedestrian attribute recognition.

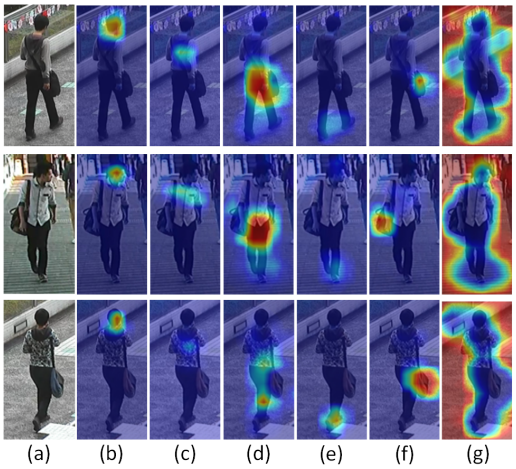

Figure 3: Visualizations of the soft-assignments in CRM. Except for the first column denoting the original image, each column indicates a soft-assignment for a graph node.

Comparisons to Prior Arts

To show the effectiveness of the proposed JLAC, we take the methods of CNN+SVM (Li et al. 2018b), ACN (Sudowe, Spitzer, and Leibe 2015), DeepMar (Li, Chen, and Huang 2015), HP-net (Liu et al. 2017), VeSPA (Sarfraz et al. 2017), JRL (Wang et al. 2017), Fusion (Li et al. 2018a), LG-Net (Liu et al. 2018), VAA (Sarafianos, Xu, and Kakadiaris 2018), GRL (Zhao et al. 2018), RA (Zhao et al. 2019), JLPLS-PAA (Tan et al. 2019b) and CoCNN (Han et al. 2019) for comparisons. The comparisons are summarized in Table 2, Table 3 and Table 4 for PETA, RAP and PA-100K datasets, respectively. The proposed method JLAC achieves new state-of-the-art performance on PETA, RAP and PA-100K datasets, with achieving the mean performance over five criteria of 85.94%, 79.07% and 84.92%, respectively. More specifically, taking the PETA dataset as an example, JLAC outperforms the most recent state-of-the-art methods, JLPLS-PAA, by 0.95% on the mean performance over five criteria. It is a promising improvement because the performance is averaging on dozens of attributes where the accuracies of some attributes are really hard to be improved due to the low resolution, occlusions, unbalanced data and so on. Some methods like VeSPA, GRL, Fusion and JLPLS-PAA exploits the external information for further improving performance, while our JLAC still achieves the best results. For example, GRL and JLPLS-PAA employ the pedestrian pose and parsing information, respectively. Some previous methods may perform well on a single metric, but our method achieves best on the overall performance. For example, on RAP and PA-100K datasets, although CoCNN can obtain higher performance on Prec, its mean performance over the

Table 5: The comparisons of GCN and LSTM.

Module	ARM	CRM
with GCN	84.95	85.03
with LSTM	84.65	84.53

given five metrics is lower than ours by 0.73% and 1.01%, respectively. The promising performance on three datasets of JLAC clearly shows the superiority of exploring attribute and contextual relations.

Further Analysis and Discussions

Visualizations of Soft-assignment We select 6 nodes in the CRM and visualize their soft-assignments on the test set of PETA dataset. As shown in Fig. 3, different nodes focus on different image regions. More specifically, from Fig. 3 (b)-(g), the nodes may focus on head, upper body, middle body, shoes, accessories and background. The soft-assignment mechanism works well although there are large variations like arbitrary human poses, different camera viewing angles and so on. Moreover, the last node aggregates the features from the background, which may help the learning of later features by separating the extraneous information from other nodes. The visualizations can qualitatively show the employed graph projection scheme can really aggregate the features from some important regions.

GCN vs. LSTM In previous work, there are some works employing LSTM (Hochreiter and Schmidhuber 1997) to capture the relations of attributes (Wang et al. 2017) or human body parts (Zhao et al. 2018). However, LSTM depends on the order of the sequential data. Thus, wang et al. (Wang et al. 2017) employ an ensemble of models and Zhao et al. (Zhao et al. 2018) carefully design the order of the inputs for achieving promising performance. To verify the effectiveness of GCN employed in our modules, we conduct the experiments of replacing the GCN with LSTM in both ARM and CRM. The comparisons of the mean performance over five criteria are shown in Table 5. GCN can achieve better performance than LSTM on both ARM and CRM, which demonstrates its effectiveness.

Qualitative Analysis The predictions of two examples on PA-100K dataset are shown in Fig. 4. The ground truth (GT) labels and the predictions of PAR and JLAC are denoted by red, green and blue colors, respectively. In those samples, JLAC can achieve better results by exploring the attribute and contextual relations. For example, for the image of the first row, PAR makes wrong predictions on "ShortSleeve" and "Back", while JLAC can well correct them by exploring the attribute and contextual relations.

Analysis on Improvements We draw the mean accuracy (mA) results on RAP dataset of the proposed JLAC and the baseline PAR as shown in Fig. 5. From Fig. 5, we find that JLAC can almost improve the accuracies on all attributes. Some attributes like "Glasses", "Muffler" and "Casual Shoes" are hardly inferred directly from the image, and their large improvements may come from the exploration of

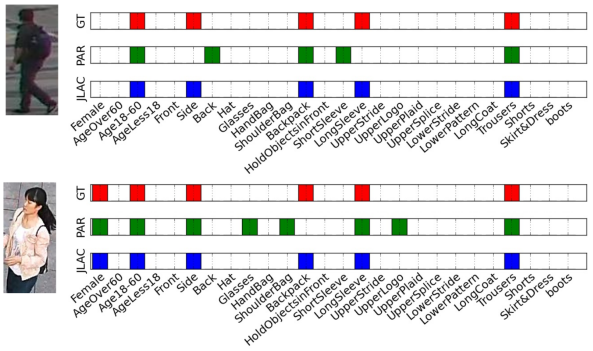

Figure 4: Examples of the predictions on PA-100K dataset.

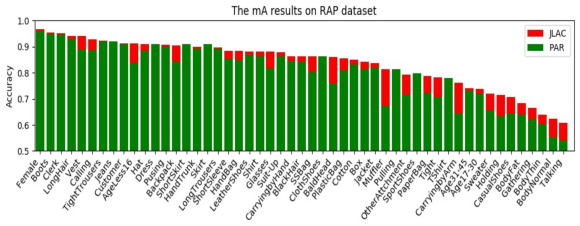

Figure 5: The mA results of all attributes on RAP dataset.

attribute relations. Moreover, the improvements on some attributes like "CarryingbyArm", "Holding" and "BodyNormal" are also very evident. Those attributes require a consideration from multiple image regions when recognizing them. Their improvements may owe to the effectiveness of exploring contextual relations.

Conclusion

To improve pedestrian attribute recognition, we have proposed a unified framework, named JLAC, which contains two novel modules, namely ARM and CRM. The attribute graph in ARM is constructed based on the attribute-specific features. The graph in CRM is produced based on a graph projection scheme by projecting the feature map into a set of nodes. In both modules, the GCN is further developed to exploit the relations. The experiments on PETA, RAP and PA-100K datasets have demonstrated that the JLAC outperforms the previous state-of-the-art methods. Moreover, we also have provided feature visualizations and a comprehensive analysis of JLAC, which can qualitatively demonstrate its effectiveness.

Acknowledgments

This work was supported by the Chinese National Natural Science Foundation Projects #61961160704, #61876179, #61872367, #61806203, Science and Technology Development Fund of Macau (No. 0008/2018/A1, 0025/2019/A1, 0019/2018/ASC, 0010/2019/AFJ, 0025/2019/AKP).

References

Arandjelovic, R.; Gronat, P.; Torii, A.; Pajdla, T.; and Sivic, J. 2016. Netvlad: Cnn architecture for weakly supervised place recognition. In *CVPR*.

Chen, Z.-M.; Wei, X.-S.; Wang, P.; and Guo, Y. 2019. Multi-label image recognition with graph convolutional networks. In *CVPR*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*.

Deng, Y.; Luo, P.; Loy, C. C.; and Tang, X. 2014. Pedestrian attribute recognition at far distance. In *ACM MM*.

Feris, R.; Bobbitt, R.; Brown, L.; and Pankanti, S. 2014. Attribute-based people search: Lessons learnt from a practical surveillance system. In *ACM MM*.

Gori, M.; Monfardini, G.; and Scarselli, F. 2005. A new model for learning in graph domains. In *IJCNN*.

Han, K.; Wang, Y.; Shu, H.; Liu, C.; Xu, C.; and Xu, C. 2019. Attribute aware pooling for pedestrian attribute recognition. In *IJCAI*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural computation*.

Ioffe, S., and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*.

Kingma, D. P., and Ba, J. 2015. Adam: A method for stochastic optimization. In *ICLR*.

Kipf, T. N., and Welling, M. 2017. Semi-supervised classification with graph convolutional networks. In *ICLR*.

Li, Y., and Gupta, A. 2018. Beyond grids: Learning graph representations for visual recognition. In *NeurIPS*.

Li, Y.; Tarlow, D.; Brockschmidt, M.; and Zemel, R. 2016. Gated graph sequence neural networks. In *ICLR*.

Li, D.; Chen, X.; Zhang, Z.; and Huang, K. 2018a. Pose guided deep model for pedestrian attribute recognition in surveillance scenarios. In *ICME*.

Li, D.; Zhang, Z.; Chen, X.; and Huang, K. 2018b. A richly annotated pedestrian dataset for person retrieval in real surveillance scenarios. *IEEE TIP*.

Li, Q.; Zhao, X.; He, R.; and Huang, K. 2019a. Pedestrian attribute recognition by joint visual-semantic reasoning and knowledge distillation. In *IJCAI*.

Li, Q.; Zhao, X.; He, R.; and Huang, K. 2019b. Visual-semantic graph reasoning for pedestrian attribute recognition. In *AAAI*.

Li, D.; Chen, X.; and Huang, K. 2015. Multi-attribute learning for pedestrian attribute recognition in surveillance scenarios. In *ACPR*.

Lin, Y.; Zheng, L.; Zheng, Z.; Wu, Y.; Hu, Z.; Yan, C.; and Yang, Y. 2019. Improving person re-identification by attribute and identity learning. *Pattern Recognition*.

Liu, X.; Zhao, H.; Tian, M.; Sheng, L.; Shao, J.; Yi, S.; Yan, J.; and Wang, X. 2017. Hydraplus-net: Attentive deep features for pedestrian analysis. In *ICCV*.

Liu, P.; Liu, X.; Yan, J.; and Shao, J. 2018. Localization guided learning for pedestrian attribute recognition. In *BMVC*.

Luo, W.; Li, Y.; Urtasun, R.; and Zemel, R. 2016. Understanding the effective receptive field in deep convolutional neural networks. In *NeurIPS*.

Sarafianos, N.; Xu, X.; and Kakadiaris, I. A. 2018. Deep imbalanced attribute classification using visual attention aggregation. In *ECCV*.

Sarfraz, M. S.; Schumann, A.; Wang, Y.; and Stiefelhagen, R. 2017. Deep view-sensitive pedestrian attribute inference in an end-to-end model. In *BMVC*.

Scarselli, F.; Gori, M.; Tsoi, A. C.; Hagenbuchner, M.; and Monfardini, G. 2009. The graph neural network model. *IEEE TNN*.

Simonyan, K., and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. In *ICLR*.

Sudowe, P.; Spitzer, H.; and Leibe, B. 2015. Person attribute recognition with a jointly-trained holistic cnn model. In *ICCVW*.

Tan, Z.; Yang, Y.; Wan, J.; Guo, G.; and Li, S. Z. 2019a. Deeply-learned hybrid representations for facial age estimation. In *IJCAI*.

Tan, Z.; Yang, Y.; Wan, J.; Hang, H.; Guo, G.; and Li, S. Z. 2019b. Attention-based pedestrian attribute analysis. *IEEE TIP*.

Velivckovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2018. Graph attention networks. In *ICLR*.

Wang, J.; Zhu, X.; Gong, S.; and Li, W. 2017. Attribute recognition by joint recurrent learning of context and correlation. In *ICCV*.

Wang, Y.; Gan, W.; Wu, W.; and Yan, J. 2019. Dynamic curriculum learning for imbalanced data classification. In *ICCV*.

Xiang, L.; Jin, X.; Ding, G.; Han, J.; and Li, L. 2019. Incremental few-shot learning for pedestrian attribute recognition. In *IJCAI*.

Yan, S.; Xiong, Y.; and Lin, D. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*.

Zhao, X.; Sang, L.; Ding, G.; Guo, Y.; and Jin, X. 2018. Grouping attribute recognition for pedestrian with joint recurrent learning. In *IJCAI*.

Zhao, X.; Sang, L.; Ding, G.; Han, J.; Di, N.; and Yan, C. 2019. Recurrent attention model for pedestrian attribute recognition. In *AAAI*.

Zheng, L.; Shen, L.; Tian, L.; Wang, S.; Wang, J.; and Tian, Q. 2015. Scalable person re-identification: A benchmark. In *ICCV*.