Drop Loss for Person Attribute Recognition with Imbalanced Noisy-Labeled Samples

Yan Yan, *Member, IEEE*, Youze Xu, Jing-Hao Xue, *Senior Member, IEEE*, Yang Lu, Hanzi Wang, *Senior Member, IEEE*, and Wentao Zhu

***Abstract*—Person attribute recognition (PAR) aims to simultaneously predict multiple attributes of a person. Existing deep learning-based PAR methods have achieved impressive performance. Unfortunately, these methods usually ignore the fact that different attributes have an imbalance in the number of noisy-labeled samples in the PAR training datasets, thus leading to suboptimal performance. To address the above problem of imbalanced noisy-labeled samples, we propose a novel and effective loss called drop loss for PAR. In the drop loss, the attributes are treated differently in an easy-to-hard way. In particular, the noisy-labeled candidates, which are identified according to their gradient norms, are dropped with a higher drop rate for the harder attribute. Such a manner adaptively alleviates the adverse effect of imbalanced noisy-labeled samples on model learning. To illustrate the effectiveness of the proposed loss, we train a simple ResNet-50 model based on the drop loss and term it DropNet. Experimental results on two representative PAR tasks (including facial attribute recognition and pedestrian attribute recognition) demonstrate that the proposed DropNet achieves comparable or better performance in terms of both balanced accuracy and classification accuracy over several state-of-the-art PAR methods.**

***Index Terms*—Person attribute recognition, imbalanced noisy-labeled samples, gradient norm, deep learning.**

I. Introduction

PERSON attributes are mid-level semantic features that can describe certain characteristics (such as gender and age) of a person in an image, and they are beneficial for high-level computer vision tasks, including face verification and recognition [1], person re-identification [2], and action recognition [3]. As an important task of multi-attribute learning, person attribute recognition (PAR) aims to simultaneously predict multiple attributes of a given person image. Although significant progress has been made during the past few years, PAR is still a challenging problem due to large person appearance variations caused by different poses, viewpoints, illuminations, etc.

Early PAR methods rely on powerful low-level features (e.g., histogram of oriented gradients (HOG) [1] and subspace learning [4]) and traditional classifiers (e.g., support vector machine (SVM) [5], [6]). For example, Kumar *et al.* [1] extract low-level image features, and then multiple SVM classifiers are separately trained for different attributes. However, the underlying relationship among attributes, which plays an important role in improving the generalization performance of PAR by sharing information across attributes, is not well exploited.

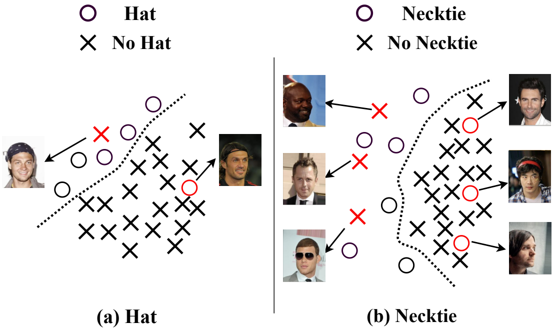

Fig. 1. An illustrative example of noisy-labeled samples for the binary attributes (a) “Hat” and (b) “Necktie” in the CelebA training dataset. The noisy-labeled samples are denoted as red circles and red crosses.

Recently, a large number of deep learning-based PAR methods [7]–[11] have been developed and have shown promising performance. These methods typically formulate the problem of predicting attributes as the problem of designing proper deep neural networks on the training datasets. Note that the samples (facial or pedestrian images) in current PAR training datasets are often collected in the wild. In these datasets, it is relatively difficult to annotate the samples with high quality due to the subjectiveness of annotators and the ambiguity of person attributes. As a result, a number of noisy-labeled samples exist.

An illustrative example of noisy-labeled samples for the binary attributes “Hat” and “Necktie” in the CelebA training dataset [12] is given in Figure 1. We can see that the number of noisy-labeled samples for the “Hat” attribute is less than that for the “Necktie” attribute. In other words, multiple attributes exhibit an imbalance in the number of noisy-labeled samples in the training datasets, mainly due to the different annotating difficulties/errors of person attributes. Noisy-labeled samples usually have a negative influence on optimizing the objective loss function, which enforces the model to focus on learning the distribution of noisy-labeled samples. As a result, the converged model is prone to have an inferior discriminative ability to classify unseen samples. Therefore, the problem of imbalanced noisy-labeled samples seriously affects the accuracy and stability of the learned model. Training a deep network

Corresponding Author: Wentao Zhu.

Y. Yan, Y. Xu, Y. Lu, H. Wang are with the Fujian Key Laboratory of Sensing and Computing for Smart City, School of Informatics, Xiamen University, Xiamen 361005, China (e-mail: xuyouze@stu.xmu.edu.cn; yanyan@xmu.edu.cn; luyang@xmu.edu.cn; hanzi.wang@xmu.edu.cn).

J.-H. Xue is with the Department of Statistical Science, University College London, London WC1E 6BT, UK (e-mail: jinghao.xue@ucl.ac.uk).

W. Zhu is with Zhejiang Lab, Hangzhou 311121, China (e-mail: wentao.zhu@zhejianglab.com).

without considering imbalanced noisy-labeled samples is not an optimal solution for PAR. The above problem, which is rarely taken into consideration in previous PAR methods, is a rewarding research problem meriting investigation.

In this paper, we propose a novel and effective loss called drop loss for PAR. To alleviate the problem of imbalanced noisy-labeled samples, the attributes are treated differently in an easy-to-hard way. In particular, the noisy-labeled candidates identified by the gradient norms are dropped with a higher drop rate for the harder attribute. In this manner, the model is adaptively learned to deal with the different numbers of noisy-labeled samples on various attributes. To illustrate the effectiveness of the proposed drop loss, we train a simple ResNet-50 model [13] based on the proposed loss and term it DropNet.

Our main contributions are summarized as follows:

- We propose a novel and simple drop loss based on gradient norms to address the problem of imbalanced noisy-labeled samples for PAR. It effectively handles the label noise and learning bias for PAR. To the best of our knowledge, this work is the first to drop noisy-labeled candidates in an easy-to-hard way to deal with unreliable multi-attribute data.
- We integrate the drop loss into the ResNet-50 model and apply it to PAR. Without bells and whistles, the model achieves state-of-the-art performance on several challenging PAR datasets (including facial attribute and pedestrian attribute datasets). This demonstrates the excellent generalization capability of the drop loss on different tasks of PAR.

The rest of the paper is organized as follows. First, we review the related work in Sec. II. Then, we present the details of the proposed drop loss in Sec. III. Next, we demonstrate the performance of the drop loss-based DropNet and compare it with several state-of-the-art PAR methods in Sec. IV. Finally, we conclude our work in Sec. V.

II. Related work

In this section, we briefly review the related work. Person attribute recognition, learning with class imbalance, and learning with noisy labels are introduced.

A. *Person Attribute Recognition*

Over the past few years, person attribute recognition (PAR), such as facial and pedestrian attribute recognition, has attracted increasing interest in a variety of applications, including face verification and recognition, action recognition, and attribute editing. For example, Kumar *et al.* [1] perform face verification based on attribute classifiers. Hu *et al.* [14] show that the face recognition performance can be greatly enhanced by incorporating predicted facial attributes into the model. Iranmanesh *et al.* [15] propose a coupled deep neural network architecture, which uses facial attributes to improve the sketch-photo recognition performance.

Early PAR methods [1], [5], [6] usually rely on handcrafted features and learn a single classifier for each attribute. Kumar *et al.* [5] extract HOG and color histograms in important functional facial regions and then train multiple SVMs for facial attribute recognition. Bourdev *et al.* [6] develop a three-level SVM method to extract high-level semantic information. The above methods usually separately train a classifier (such as SVM) for each attribute. Therefore, the underlying correlation among attributes is not effectively exploited.

Due to its powerful feature representations, deep learning has shown impressive improvements over traditional methods [16]–[18]. Most state-of-the-art PAR methods are based on deep convolutional neural networks (CNNs). Liu *et al.* [12] design a deep CNN consisting of LNet and ANet, which respectively locate the face and predict facial attributes. Abdulnabi *et al.* [19] propose a joint multi-task learning method to share the information among attributes for clothing attribute prediction. Hand and Chellappa [9] adopt a multi-task learning CNN method based on a grouping scheme to classify facial attributes. Han *et al.* [20] design a deep multi-task learning CNN method (DMTL) that learns the shared features for all the attributes and the category-specific features for heterogeneous attributes. Li *et al.* [21] propose a landmark-free facial attribute prediction method without relying on landmark annotations. Li *et al.* [22] introduce pedestrian structure knowledge into pedestrian attribute recognition and propose a pose-guided deep model to predict pedestrian attributes. Tan *et al.* [23] develop a pedestrian attribute analysis method based on three attention mechanisms: parsing attention, label attention, and spatial attention.

Recently, many efforts have been made in terms of architecture design and attribute grouping. He *et al.* [24] propose a sharing mechanism that is capable of exploiting the relationship among different person attributes. Huang *et al.* [25] propose an efficient greedy neural architecture search approach (GNAS) to automatically discover the optimal tree-like architecture for PAR. Lin *et al.* [11] develop a new multi-task network for facial attribute analysis. They design task-oriented feature-fused blocks, where each task learns effective feature combinations for classification. Shu *et al.* [26] learn the spatial-semantic relationship for facial attribute recognition with limited labeled data, where three auxiliary tasks are jointly designed to obtain a powerful pretrained model. Lingenfelter and Hand [27] suggest improvements to model evaluation for facial attribute recognition. Jia *et al.* [28] construct a spatial and semantic consistency framework, which models the inter-image relation of the same attribute, for pedestrian attribute recognition. Aslan *et al.* [29] develop a novel multimodal method based on three subnetworks (e.g., ResNet, VGGish, and ELMo), for the estimation of apparent personality traits. Moreover, they leverage additional long short-term memory (LSTM) layers to exploit temporal information.

Unlike the above methods, we investigate the problem of imbalanced noisy-labeled samples, which concerns the influence of noisy-labeled samples ubiquitously existing in the PAR training datasets on model learning and has not drawn much attention for PAR so far.

B. *Learning with Class Imbalance*

Imbalanced learning is a long-standing problem in machine learning and computer vision. Generally, existing methods to deal with class imbalanced data can be divided into three categories [30]: data-level methods, algorithm-level methods, and hybrid methods.

The data-level methods either oversample the minority class or undersample the majority class to balance the class distribution [31]. However, it is not trivial to balance all the attributes at the data-level for the PAR problem. Algorithm-level methods often rely on cost-sensitive learning. In [32], [33], new loss functions are introduced to enable the training process to focus on classifying the minority samples. In [34], the samples from the minority class are assigned higher misclassification costs than those from the majority class. Finally, the hybrid methods combine the advantages of data-level and algorithm-level methods. Huang *et al.* [35] propose the CLMLE method, which uses the clustering technique to capture the local distribution of clusters for each class, in order to preserve the same-class locality and increase the inter-class discrimination for imbalanced image data. Dong *et al.* [36] develop a class rectification loss and employ a hard sample mining technique to discover the latent boundaries of individual classes.

Different from the traditional problem of imbalanced learning that generally deals with class imbalanced data, we focus on the problem of imbalanced noisy-labeled samples that may also significantly decrease the accuracy of the model.

C. Learning with Noisy Labels

Real-world datasets usually contain noisy-labeled samples [37]–[39]. Recently, learning with noisy labels has received increasing attention in the computer vision community. Current methods can be classified into three categories [40]: label noise-robust methods, label noise-tolerant methods, and data cleansing methods.

Representative label noise-robust methods are ensemble methods, such as bagging [41]. However, these methods do not explicitly take into account the label noise. The label noise-tolerant methods model the label noise during learning. For example, Xiao *et al.* [42] propose a method to characterize the relationships between images, truth labels, and noisy labels by using a probabilistic graphical model for image classification. But their method requires a small set of clean labels. Goldberger *et al.* [43] propose to estimate the connections between truth labels and noisy labels by using an adaptation softmax layer. Zhang *et al.* [44] develop a generalized cross-entropy loss for classification with noisy labels. Ding *et al.* [45] propose a deep confidence network (DECODE) to address the problem of noisy-labeled samples, where a confidence evaluation module is introduced to determine the confidence of a noisy-labeled sample. Hence, the model can pay more attention to the high confidence data. Recently, Zeng *et al.* [46] explore inconsistently labeled samples among different facial expression recognition datasets and develop a method to discover the latent truth from inconsistent pseudo-labels and input facial images. Wang *et al.* [47] propose to suppress the uncertainties (caused by incorrect/noisy annotations) in facial expression images to learn robust features.

Data cleansing methods improve the quality of training data by removing noisy-labeled samples. Zhang *et al.* [48] propose to detect both noisy-labeled samples and hard training samples by using a small group of trusted data. Speth *et al.* [39] design a multi-label Siamese network to project the image onto a lower-dimensional space, and then perform attribute verification between a candidate sample and a set of representative samples to identify noisy-labeled samples. Huang *et al.* [49] claim that the probability of a sample being a noisy-labeled sample is closely related to the normalized average loss for this sample. But this method depends on a parameter k, which is used to determine the proportion of noisy-labeled samples in the dataset, by manually verifying a small number of randomly selected samples. Note that the above methods need to manually select a clean dataset (consisting of correctly labeled samples) to detect the noisy-labeled samples.

Our proposed method is also a data cleansing method. However, in contrast to previous methods [39], [49], we propose to identify the noisy-labeled samples in a dynamic and progressive manner according to their gradient norms. The proposed method not only effectively handles PAR with noisy labels, but also does not require manual selection of a clean dataset for model learning.

III. Methodology

In this section, a novel and effective loss is developed to address the problem of imbalanced noisy-labeled samples for PAR. We introduce the preliminary knowledge, noisy-labeled sample modeling, and problem description in Sec. III-A, Sec. III-B, and Sec. III-C, respectively. Then, we describe the proposed drop loss in detail in Sec. III-D. Finally, we discuss our proposed method in Sec. III-E.

A. Preliminary Knowledge

Given a training set $\mathcal{T}$ and corresponding labels $\mathcal{Y}$ with P training images and M attributes, where $\mathcal{T} = \{\mathbf{I}_i\}_{i=1}^{P}$ and $\mathbf{I}_i$ denotes the i-th image in the training set. $\mathcal{Y} = \{y_{i,j}\}_{i=1,j=1}^{P,M}$, where $y_{i,j} \in \{0,1\}$. Here, $y_{i,j} = 1$ represents that the i-th image is annotated to have the j-th attribute, and $y_{i,j} = 0$ otherwise. During the prediction stage, given a test image $\mathbf{I}_t$, the goal is to predict an attribute vector $\mathbf{y} \in \{0,1\}^M$ by using the trained model.

Let $x_{i,j}$ be the output of the model for the j-th attribute of the i-th image $\mathbf{I}_i$. $p_{i,j}$ denotes the probability predicted by the model with respect to $x_{i,j}$, and is usually computed according to the sigmoid function. For the j-th attribute of $\mathbf{I}_i$, we can define the binary cross-entropy (CE) loss as

$$\mathcal{L}_{ce}(x_{i,j}) = -[y_{i,j}\log(p_{i,j}) + (1-y_{i,j})\log(1-p_{i,j})]. \quad (1)$$

Meanwhile, the gradient of $\mathcal{L}_{ce}(x_{i,j})$ with respect to $x_{i,j}$ can be derived as

$$\begin{aligned}\frac{\partial \mathcal{L}_{ce}(x_{i,j})}{\partial x_{i,j}} =& y_{i,j}(p_{i,j}-1) + (1-y_{i,j})p_{i,j} \\ =& p_{i,j} - y_{i,j}.\end{aligned} \quad (2)$$

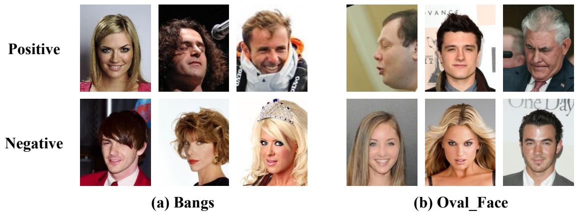

Fig. 2. Examples of noisy-labeled samples for the (a) “Bangs” and (b) “Oval_Face” attributes in the CelebA training dataset. The upper row and the lower row correspond to the positive samples and the negative samples in the training set, respectively. All these samples are identified as noisy-labeled samples by our converged model.

Similar to [33], we define the gradient norm with respect to $x_{i,j}$ for the j-th attribute of $\mathbf{I}_i$ as follows:

$$g(x_{i,j}) = |p_{i,j} - y_{i,j}| = \begin{cases} 1 - p_{i,j}, & \text{if } y_{i,j} = 1, \\ p_{i,j}, & \text{if } y_{i,j} = 0, \end{cases} \tag{3}$$

where $g(x_{i,j}) \in [0, 1]$ is the absolute value of the gradient $\partial \mathcal{L}_{ce}(x_{i,j})/\partial x_{i,j}$. The value of $g_{i,j}$ represents the discrepancy between the predicted probability and the ground-truth label for the j-th attribute of $\mathbf{I}_i$. The larger the value of $g_{i,j}$ is, the less accurate the predicted probability is. When the value of the gradient norm is 0.5, the predicted probability of a sample is 0.5. In this case, it is difficult to determine the label of this sample (note that the ground-truth label of each sample is either 0 or 1 for an attribute). This indicates that the sample is on the decision boundary of the classifier.

B. Noisy-Labeled Sample Modeling

As discussed in Sec III-A, the gradient norm indicates the difficulty of a trained model in predicting an attribute. Therefore, it can be effectively used to select noisy-labeled samples.

More formally, given a batch of n training samples $\mathcal{I}^b = \{\mathbf{d}_i^b\}_{i=1}^n$ and their corresponding labels $\mathcal{Y}^b = \{y_{i,j}^b\}_{i=1,j=1}^{n,M}$ at epoch t, the noisy-labeled candidates for the j-th attribute are defined as

$$\mathcal{O}_j^b = \{\mathbf{d}_i^b | g(x_{i,j}^b) \geq \alpha^t\}, \tag{4}$$

where $x_{i,j}^b$ represents the output of the model for the j-th attribute of $\mathbf{d}_i^b$; $g(x_{i,j}^b)$ denotes the gradient norm with respect to $x_{i,j}^b$; $\mathcal{O}_j^b$ is the noisy-labeled candidate set consisting of the image samples whose gradient norms are larger than α^t for the j-th attribute; and α^t is a threshold to determine the noisy-labeled candidate, which is defined as

$$\alpha^t = 1 - \lambda \cdot \min(\frac{t}{T}, 1), \tag{5}$$

where $\lambda \in [0, 1]$ is a tuning parameter. The value of α^t decreases quickly at the first T epochs until reaching $1 - \lambda$. Such a strategy can prevent the model from overfitting to noisy-labeled samples.

During the initial training process, the noisy-labeled candidate set may contain some useful hard training samples with large gradient norms. As training proceeds, the model becomes more accurate, and thus more noisy-labeled samples can be selected. When training ends, the noisy-labeled candidate set is mainly composed of noisy-labeled samples. Therefore, the noisy-labeled samples are identified in a dynamic and progressive manner. In this way, the influence of noisy-labeled samples is gradually alleviated during training. Note that the above learning procedure is different from curriculum learning [50], where the training process pays attention to easy samples at the beginning and gradually focuses on difficult samples in subsequent training.

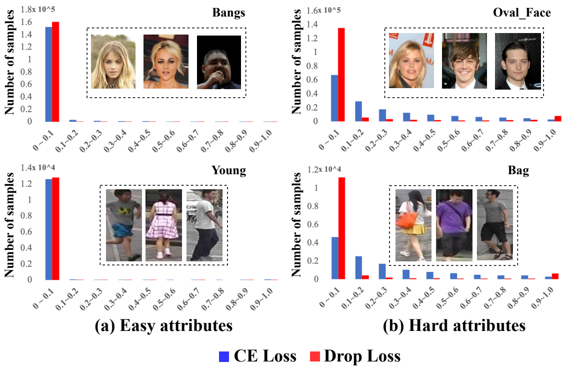

Fig. 3. The histograms of gradient norms from the converged ResNet-50 models by using the CE loss (in blue) and our drop loss (in red) on the CelebA (upper row) and Market-1501 attribute (lower row) training datasets, respectively. The x-axis value and y-axis value of each histogram represent the bin of gradient norms and its corresponding number of samples. (a) The easy attributes with a small number of noisy-labeled samples, and (b) the hard attributes with a relatively large number of noisy-labeled samples.

Fig. 2 shows several noisy-labeled samples identified by our converged model for the “Bangs” and “Oval_Face” attributes in the CelebA training dataset [12]. We find that these noisy-labeled samples are unreliable samples.

It is worth pointing out that the noisy-labeled samples belong to the hard training samples since the converged model cannot accurately predict the attributes of these samples. Normally, hard training samples play a critical role in learning a discriminative classifier [5], [6]. However, the noisy-labeled samples are totally different from the truly useful hard training samples. As illustrated in Fig. 1, the noisy-labeled samples refer to the unreliable samples (e.g., the red points), while the useful hard training samples are the ones near the decision hyperplane (e.g., the points near the dotted lines). In fact, the gradient norms (see Eq. (4)) of these useful hard training samples are near 0.5 (which indicates that the positive and negative samples are not easily distinguishable). In other words, these useful hard training samples are near the decision boundary. In general, samples near the decision boundary of a classifier have a larger influence on the final performance than those far apart from it [51]. In contrast, the gradient norms of noisy-labeled samples are large. The noisy-labeled samples have a negative effect on learning accurate deep models. If the model is enforced to discriminate the noisy-labeled samples, it fails to learn the intrinsic distribution of attributes. Therefore, by dropping these noisy-labeled samples from hard training samples, the model effectively reduces the risk of overfitting.

C. Problem Description

To illustrate the problem of imbalanced noisy-labeled samples, Fig. 3 shows the histograms of gradient norms (HGN) corresponding to four attributes based on the converged ResNet-50 models by using the CE loss and our proposed drop loss in the CelebA training dataset and the Market-1501 attribute training dataset [2]. We can observe that the HGNs corresponding to the "Bangs" and "Young" attributes (Fig. 3(a)) are significantly different from those corresponding to the "Oval_face" and "Bag" attributes (Fig. 3(b)).

Specifically, for the attributes in Fig. 3(a), the number of samples whose gradient norms are small is extremely large, and these samples account for a large proportion by using the CE loss. Such attributes can be regarded as *easy* attributes since the gradient norms of a large number of samples are close to zero. In other words, most samples for easy attributes are easily classified. Therefore, the training losses of these easy attributes are small. Meanwhile, the number of noisy-labeled samples (having large gradient norms) for these easy attributes is also trivial. In contrast, for the attributes in Fig. 3(b), the number of misclassified samples, whose gradient norms are above 0.5, is large by using the CE loss. Such attributes can be regarded as *hard* attributes, where many samples for these attributes are incorrectly classified. The number of noisy-labeled samples for these hard attributes is not trivial. Hence, the training losses of these hard attributes are large.

Generally, we observe that easy attributes have a small number of noisy-labeled samples, while hard attributes have a relatively large number of noisy-labeled samples. If we force the model to learn these noisy-labeled samples, the model tends to be inaccurate (i.e., the classification performance of unseen samples decreases due to overfitting of the training data). Therefore, how to deal with imbalanced noisy-labeled samples among attributes is an important issue for PAR.

D. Drop Loss

To effectively address the above problem, we propose a novel drop loss. We process the training data in a batch-wise manner, where each batch consists of samples from the training set.

One straightforward way to deal with imbalanced noisy-labeled samples is to simply drop all the noisy-labeled samples. However, the identification of noisy-labeled samples is not accurate at the initial iterations of the training. Therefore, we treat the attributes differently in an easy-to-hard way, where we drop noisy-labeled candidates according to different drop rates for multiple attributes.

Suppose that the noisy-labeled candidate set for the j-th attribute is denoted as $\mathcal{O}_j^b = \{\mathbf{d}_1^b, \cdots, \mathbf{d}_{H_j}^b\}$ (obtained by Eq. (4)) and the corresponding gradient norms are denoted as $\mathcal{G}_j^b = \{g_{1,j}^b \cdots, g_{H_j,j}^b\}$, where H_j is the number of noisy-labeled candidates for the j-th attribute and $g_{i,j}^b$ represents the gradient norm of the i-th sample $\mathbf{d}_i^b$.

First, we obtain the average gradient norm G_j^b of the batch data for the j-th attribute as

$$G_j^b = \frac{1}{n}\sum_{i=1}^{n} g(x_{i,j}^b). \tag{6}$$

The values of the average gradient norm vary greatly for different attributes. Therefore, we normalize the values of the average gradient norm to $[0, 1]$ according to min-max normalization, which is formulated as

$$DR_j^b = \frac{G_j^b - \min\limits_{j=1,\cdots,M}\{G_j^b\}}{\max\limits_{j=1,\cdots,M}\{G_j^b\} - \min\limits_{j=1,\cdots,M}\{G_j^b\}}, \tag{7}$$

where DR_j^b is defined as the drop rate for the j-th attribute.

Then, the noisy-labeled candidates for an attribute are dropped based on the drop rate of an attribute. We drop noisy-labeled candidates at a higher rate for harder attributes that have more noisy-labeled samples. That is, the more difficult an attribute is, the higher its drop rate is.

More specifically, we sort all the elements in $\mathcal{G}_j^b$ in descending order to obtain the sorted set $\hat{\mathcal{O}}_j^b = \{\mathbf{d}_{\mu_1}^b, \cdots, \mathbf{d}_{\mu_{H_j}}^b\}$, where the permutation $\{\mu_1, \cdots, \mu_{H_j}\}$ is obtained such that $g_{\mu_1,j}^b \geq \cdots, \geq g_{\mu_{H_j},j}^b$. Hence, the noisy-labeled sample drop set is obtained as

$$\mathcal{O}_j^{b*} = \{\mathbf{d}_{\mu_k}^b, \forall k \in [1, \beta \cdot DR_j^b \cdot H_j]\}, \tag{8}$$

where $\beta \in [0, 1]$ is the scaling parameter, and $\mathbf{d}_{\mu_k}^b$ denotes the k-th sample chosen from the noisy-labeled candidate set $\hat{\mathcal{O}}_j^b$ for the j-th attribute.

According to Eq. (8), the noisy-labeled samples for multiple attributes are dropped with adaptive drop rates. The drop rate for each attribute is computed based on the gradient norms of samples in the noisy-labeled candidate set. In other words, we first analyze the statistical information based on the noisy-labeled candidate set and then adaptively drop noisy-labeled samples according to this information. Such a way can be viewed as a two-stage sample filtering method for data cleansing.

Finally, the batch training data for the j-th attribute are composed of $\mathcal{I}_j^{b*} = \mathcal{I}^b \setminus \mathcal{O}_j^{b*}$. Accordingly, the drop loss for the j-th attribute is then formulated as

$$\mathcal{L}_j^{b*} = \frac{1}{|\mathcal{I}_j^{b*}|}\sum_{i=1}^{|\mathcal{I}_j^{b*}|} \mathcal{L}_{ce}(x_{i,j}^{b*}), \tag{9}$$

where $|\mathcal{I}_j^{b*}|$ denotes the number of training samples for the j-th attribute in a batch and $x_{i,j}^{b*}$ represents the output of the trained deep model for the j-th attribute of the i-th image $\mathbf{I}_i^{b*}$ in $\mathcal{I}_j^{b*}$.

Therefore, the drop loss for all the attributes in a batch is formulated as

$$\mathcal{L}_{drop} = \sum_{j=1}^{M} \mathcal{L}_j^{b*}. \tag{10}$$

From Fig. 3, we can see that our drop loss (based on Eq. (10)) can effectively alleviate the problem of imbalanced noisy-labeled samples. For both easy and hard attributes, the samples whose gradient norms are small account for a large proportion. The trained model based on the drop loss accurately classifies the most reliable training samples, and achieves much better performance than the model based on the CE loss. This can be ascribed to the fact that the attributes are treated differently in an easy-to-hard way, where the noisy-labeled candidates for each attribute are dropped according to the corresponding drop rate. In other words, the drop loss effectively improves the performance of the trained model for classifying reliable training samples.

We integrate the proposed drop loss into a simple ResNet-50 model [13] (which is widely used as the backbone network in state-of-the-art PAR methods [2], [24]) and term this model DropNet. The overall training procedure of DropNet is given in the supplementary material.

E. Discussions

Our proposed method and the classical downsampling tricks used in cost-sensitive classification share some similarities in terms of removing samples. However, the two methods are significantly different. Traditional cost-sensitive classification methods [52]–[54] address the problem of imbalanced learning, where the numbers of samples in different classes are imbalanced. These methods mainly downsample the majority class (or upsample the minority class) based on the number of samples in the class. In contrast, we focus on the problem of imbalanced noisy-labeled samples (i.e., the numbers of noisy-labeled samples are imbalanced in different attributes). This is a critical but ignored problem in PAR, where each attribute has the same number of training samples. We take advantage of the gradient norm to select and drop noisy-labeled samples for each attribute. To address the imbalance in the number of noisy-labeled samples in the training set, more noisy-labeled samples will be dropped for the harder attributes. Note that we cannot guarantee that the remaining samples are balanced in terms of class data distribution for each attribute (which is another issue to be solved in the future).

IV. Experiments

In this section, we perform experiments on two representative PAR tasks (including facial attribute recognition and pedestrian attribute recognition).

A. Datasets and Evaluation Metrics

We conducted experiments on three representative facial attribute datasets (i.e., CelebA, LFWA, and MAAD-Face) and two representative pedestrian attribute datasets (i.e., Market-1501 attribute and DukeMTMC attribute).

1) CelebA: CelebA [12] is a large-scale facial attribute dataset that contains 202,599 celebrity images of more than 10K identities. Each image is annotated with 40 binary attributes. For a fair comparison with other state-of-the-art methods, we follow the protocol provided in [12], where 162,770, 19,867, and 19,962 images are used for training, validation, and testing, respectively.

2) LFWA: LFWA [12] is another unconstrained facial attribute dataset, where the facial images are collected from the LFW dataset [55]. LFWA contains 13,143 images of 5,749 identities and provides the same 40 attribute annotations as CelebA. We follow the protocol provided in [12], which uses 6,263 images for training and 6,880 images for testing. Since no official split of the training set and the validation set is provided, we use the first 5,000 images in the training set for training and the rest 1,263 images for validation, as done in [25].

3) MAAD-Face: MAAD-Face [56] is a newly-released facial attribute dataset based on VGGFace2 [57]. MAAD-Face consists of 123.9M attribute annotations of 47 different binary attributes. It provides 15 and 137 times more labels than CelebA and LFW, respectively. The attribute annotations are of high quality. We manually select 200,000 images for training, 20,000 for validation and 20,000 for testing.

4) Market-1501 Attribute: The Market-1501 attribute dataset [2] is an extension of the Market-1501 dataset [58] with pedestrian attribute annotations. The dataset contains 32,688 images of 1,501 identities. It provides 12 different types of annotated attributes, including 9 binary attributes (such as hat, hair, gender, and sleeve length) and 3 multi-class attributes (age, colors of upper, and lower body clothing). We follow the protocol provided in [2], which uses 751 identities with 19,732 images for training and 750 identities with 13,328 images for testing. We use the first 16,000 images in the training set for training and the rest 3,732 images for validation.

5) DukeMTMC Attribute: The DukeMTMC attribute dataset [2] is a subset of the DukeMTMC dataset [59]. The dataset contains 34,183 images of 1,812 identities. It is annotated with 8 binary pedestrian attributes (such as wearing a hat and wearing boots) and 2 multi-class attributes (colors of upper and lower body clothing). We follow the protocol provided in [2], which uses 702 identities with 16,522 images for training, 2,228 images for validation, and 17,661 images for testing.

To evaluate the effectiveness and generalization capability of our method to handle imbalanced noisy-labeled samples, we report two evaluation metrics as follows.

We measure the classification performance by using the average accuracy, which is the most frequently used metric to evaluate the classification performance. The average accuracy (abbreviated as *acc*) can be formulated as

$$acc = \frac{1}{M}\sum_{j=1}^{M}\frac{TP_j + TN_j}{N}, \tag{11}$$

where TP_j and TN_j denote the numbers of true positive samples and true negative samples for the j-th attribute, respectively, and N denotes the total number of samples. $\frac{(TP_j+TN_j)}{N}$ represents the classification accuracy for the j-th attribute.

However, the average accuracy may not accurately reflect the performance of the model, when the class distribution of an attribute is extremely imbalanced. In particular, the imbalance ratios between the minority classes and the majority classes in the PAR datasets are extremely large for many attributes [35], [36], [60]. Therefore, we also employ another popular evaluation metric, called balanced accuracy [30], which is commonly used for imbalanced data. In addition, to evaluate the performance of removing noisy-labeled samples, we also use the label F1-score [61] for comparison.

The average balanced accuracy (abbreviated as *bal-acc*), can be formulated as

$$bal\text{-}acc = \frac{1}{2M}\sum_{j=1}^{M}\left(\frac{TP_j}{TP_j + FN_j} + \frac{TN_j}{TN_j + FP_j}\right), \tag{12}$$

TABLE I
FACIAL ATTRIBUTE RECOGNITION ON THE CELEBA DATASET. THE RESULTS OF TRIPLET-KNN, PANDA, LMLE, CRL-I AND CLMLE ARE CITED FROM [60]. METRIC: BALANCED ACCURACY (%). THE BEST RESULTS ARE BOLDFACED.

Attributes Methods	Attractive	Mouth_Open	Smiling	Lipstick	High_Cheekbones	Male	Heavy_Makeup	Wavy_Hair	Oval_Face	Pointy_Nose	Arched_Eyebrows	Black_Hair	Big_Lips	Big_Nose	Young	Straight_Hair	Brown_Hair	Bags_Under_Eyes	Earrings	No_Beard	
Triplet-kNN [66]	83	92	92	91	86	91	88	77	61	61	73	82	55	68	75	63	76	63	69	82	
PANDA [67]	85	93	98	97	89	**99**	95	78	66	67	77	84	56	72	78	66	85	67	77	87	
LNets+ANet [12]	87	96	97	95	89	**99**	96	81	67	69	76	90	57	78	84	69	83	70	83	93	
Down-sampling [68]	79	93	92	93	86	98	90	79	65	64	77	84	60	73	79	71	72	78	83	91	
Over-sampling [68]	83	94	93	94	88	98	92	82	67	68	81	88	63	76	85	76	78	84	86	94	
Cost-sensitive [52]	82	94	93	94	87	98	91	81	67	67	80	87	62	75	84	74	76	81	84	94	
MOON [8]	82	93	93	93	88	98	91	82	68	70	81	88	66	77	85	78	81	85	87	95	
LMLE [35]	88	96	**99**	**99**	92	**99**	**98**	83	68	72	79	92	60	80	87	73	87	73	83	96	
CRL-I [36]	83	95	93	94	89	96	84	79	66	73	80	90	68	80	84	73	86	80	83	94	
GHM-C [33]	80	93	93	93	86	98	91	77	59	59	74	83	56	68	83	65	65	75	81	93	
CLMLE [60]	**90**	**97**	**99**	98	**94**	**99**	**98**	**87**	**72**	**78**	**86**	**95**	66	**85**	**90**	80	**89**	82	86	**98**	
GCEL [44]	81	94	93	93	87	98	91	84	69	69	79	86	67	76	85	75	76	80	85	93	
SCN [47]	83	94	93	94	88	98	91	85	72	72	81	88	70	77	86	78	78	81	86	94	
Baseline	82	94	93	93	87	98	90	80	65	65	78	85	60	73	81	70	72	79	83	92	
DropNet	85	94	93	95	88	98	92	85	71	74	84	90	**70**	81	86	**82**	84	**86**	**89**	97	

Attributes Methods	Bangs	Blond_Hair	Bushy_Eyebrows	Necklace	Narrow_Eyes	5_Shadow	Receding_Hairline	Necktie	Eyeglasses	Rosy_Cheeks	Goatee	Chubby	Sideburns	Blurry	Hat	Double_Chin	Pale_Skin	Gray_Hair	Mustache	Bald	Average
Triplet-kNN [66]	81	81	68	50	47	66	60	73	82	64	73	64	71	43	84	60	63	72	57	75	71.55
PANDA [67]	92	91	74	51	51	76	67	85	88	68	84	65	81	50	90	64	69	79	63	74	76.95
LNets+ANet [12]	90	90	82	59	57	81	70	79	95	76	86	70	79	56	90	68	77	85	61	73	79.58
Down-sampling [68]	91	89	79	58	62	83	72	85	98	77	85	75	86	75	93	72	77	84	72	89	80.26
Over-sampling [68]	94	92	85	72	71	88	82	90	**99**	88	94	83	92	86	96	81	88	90	87	95	85.54
Cost-sensitive [52]	92	91	81	68	66	84	75	87	98	79	87	76	88	78	94	74	80	85	74	90	82.42
MOON [8]	94	93	86	74	74	89	84	91	**99**	88	95	85	92	87	97	84	89	91	90	95	86.51
LMLE [35]	98	**99**	82	59	59	82	76	90	98	78	95	79	88	59	**99**	74	80	91	73	90	83.83
CRL-I [36]	95	95	84	74	72	90	87	88	96	88	96	87	92	85	98	**89**	92	95	94	**97**	86.60
GHM-C [33]	90	88	75	59	59	81	70	86	98	74	86	72	86	74	94	72	74	81	72	86	78.70
CLMLE [60]	**99**	**99**	**88**	69	71	91	82	**96**	**99**	86	**98**	85	94	72	**99**	87	**94**	**96**	82	95	88.78
GCEL [44]	93	91	85	73	77	86	81	90	**99**	84	85	80	88	81	95	80	82	85	81	88	84.12
SCN [47]	93	92	86	76	**80**	87	82	90	**99**	86	86	80	90	85	95	83	87	86	82	89	85.49
Baseline	90	88	79	64	63	82	71	85	98	75	84	73	84	73	94	70	75	82	70	87	80.20
DropNet	96	94	**88**	**81**	**79**	**92**	**88**	95	**99**	**91**	96	**90**	**96**	**92**	97	**89**	93	95	**95**	**97**	**89.18**

where FN_j and FP_j denote the numbers of false negative samples and false positive samples for the j-th attribute, respectively. $\frac{1}{2}(\frac{TP_j}{TP_j+FN_j}+\frac{TN_j}{TN_j+FP_j})$ represents the balanced accuracy for the j-th attribute. Compared with acc, $bal\text{-}acc$ is more sensitive to the minority class.

B. Implementation Details

In our experiments, we use ResNet-50 [13] as the backbone network. Moreover, we appropriately modify the backbone ResNet-50, where the last fully-connected layer is reduced to have M neurons (here, M is the number of attributes in the dataset).

The backbone ResNet-50 is implemented based on PyTorch [62] and initialized from the ImageNet pretrained model [63]. We use the Adam optimizer [64], where the base learning rate is initialized to 0.001 with a linear warm-up [65] in the first 10 epochs, decayed to 0.0001 after 10 epochs, and further decayed to 0.00001 after 30 epochs. The epochs T is Eq. (5) is set to 10. The parameters λ and β in Eqs. (5) and (8) are set to 0.30 and 1.00, respectively. The influence of λ and β on the final performance is discussed in Sec. IV-E.

For both facial attribute recognition and pedestrian attribute recognition, we keep the sizes of the original images in the datasets. Our DropNet trained on the CelebA and MAAD-Face datasets converges after 60 epochs, and the training process takes approximately 10 hours with one NVIDIA GeForce GTX 2080Ti GPU. For the LFWA, Market-1501 attribute, and DukeMTMC attribute datasets, our DropNet converges after 50 epochs, and the training process takes approximately half an hour with the same GPU. For training all the models, the batch size is set to 128.

C. Comparisons on Facial Attribute Recognition

In this section, we compare the proposed DropNet with several state-of-the-art methods on the task of facial attribute recognition.

1) Competitors: We compare DropNet with thirteen state-of-the-art methods, including Triplet-KNN [66], PANDA [67],

TABLE II
FACIAL ATTRIBUTE RECOGNITION ON THE LFWA TEST DATASET. METRIC: AVERAGE BALANCED ACCURACY (%). THE BEST RESULTS ARE BOLDFACED.

Methods	*bal-acc*
LNets+ANet [12]	75.46
Downsampling [68]	78.14
Oversampling [68]	79.25
Cost-sensitive [52]	79.16
MOON [8]	80.53
GHM-C [33]	77.41
GCEL [44]	80.79
SCN [47]	82.15
Baseline	78.06
DropNet	**83.25**

LNets+ANet [12], Downsampling [68], Oversampling [68], Cost-sensitive [52], MOON [8], LMLE [35], CRL-I [36], GHM-C [33], CLMLE [60], GCEL [44], and SCN [47]. The baseline method that adopts the CE loss is also used for evaluation. We choose these competing methods because Triplet-KNN, PANDA, and LNets+ANet are representative facial attribute recognition methods. The downsampling, over-sampling, and cost-sensitive methods are typical imbalanced learning methods. MOON, LMLE, CRL-I, GHM-C, and CLMLE are state-of-the-art imbalanced learning methods that are designed to address the problem of imbalanced class data distribution in PAR. We also choose two label noise-tolerant methods (i.e., GCEL and SCN), which are specifically proposed to learn robust CNNs with noisy-labeled samples for the multi-class classification problem. In our implementation, we formulate the multi-label learning problem as a number of binary classification problems, where GCEL and SCN can be employed. Note that the data cleansing methods (such as [39], [48], [49]) are not taken for comparison since the clean dataset is not provided in the evaluation protocol.

For fair comparisons, all the imbalanced learning and label noise-tolerant methods are trained based on the same ResNet-50 network architecture. All models are trained on the same training set and evaluated on the same test set.

2) Overall Evaluation: The balanced accuracy obtained by all the competing methods on the CelebA test dataset is reported in Table I.

Our proposed DropNet significantly outperforms LNets+ANet and the baseline method, which shows the effectiveness of the proposed method for addressing imbalanced noisy-labeled samples. Note that the traditional down-sampling, over-sampling and cost-sensitive methods have much worse performance than the proposed DropNet (approximately 8.92%, 3.64%, and 6.76% decreases in terms of average balanced accuracy). This is mainly because of the negative impact of noisy-labeled samples on the final performance. GHM-C addresses the class imbalance problem for object detection by reweighting training samples according to the gradient norms based on the CE loss in one stage. However, it does not perform well in the PAR dataset. This is because the gradient norm distributions of attributes in the PAR dataset are significantly different from those in the object detection dataset. Therefore, the weighting scheme used in GHM-C may not be suitable for the PAR problem.

It is worth pointing out that some imbalanced learning methods (such as LMLE, CRL-I, and CLMLE) and our proposed method work on different aspects of multi-attribute learning. LMLE, CRL-I, and CLMLE focus on the problem of an imbalanced class data distribution (i.e., the imbalance ratios between the majority and minority classes are large for some attributes), while our DropNet alleviates the problem of imbalanced noisy-labeled samples over attributes. All of these methods are beneficial for improving the PAR performance. Compared with LMLE, CRL-I, and CLMLE, DropNet consistently achieves better performance in terms of average balanced accuracy. Moreover, a major technical difference between these imbalanced-learning methods and our DropNet can be described as follows. LMLE, CRL-I, and CLMLE construct informative minibatch samples before CNN training. Specifically, LMLE requires a computationally expensive data preprocessing step (including clustering and quintuplet construction); CRL-I constructs a set of informative triplets and combines the CE loss and the class rectification loss to rectify the class distribution bias; and CLMLE uses the clustering technique to capture the local distribution of clusters for each class and then repeatedly constructs the minibatches from the clusters. In contrast, DropNet is based on the simple drop loss without constructing computationally expensive triplets or performing clustering for each class.

TABLE III
FACIAL ATTRIBUTE RECOGNITION ON THE CELEBA AND LFWA TEST DATASETS. METRIC: AVERAGE ACCURACY (%). THE BEST RESULTS ARE BOLDFACED.

Methods	*acc*	
	CelebA	LFWA
PANDA [67]	85.43	81.03
LNets+ANet [12]	87.33	83.50
MOON [8]	90.94	85.82
Adaptive Weight [24]	**91.80**	-
MCNN-AUX [9]	91.29	86.31
GNAS [25]	91.63	86.37
GCEL [44]	90.57	84.52
SCN [47]	90.45	85.19
Baseline	91.08	84.79
DropNet	91.70	**86.52**

3) Further Results: To further evaluate the proposed DropNet on facial attribute recognition, we also show the average balanced accuracy obtained by several competing methods on the LFWA test dataset, as given in Table II. Because the source codes of LMLE, CRL-I, and CLMLE are not publicly available, we do not compare our method with these methods on LFWA, Market-1501 attribute, and DukeMTMC attribute. We can observe that the proposed DropNet obtains the best average balanced accuracy (83.25%) among all the competing methods. This can be ascribed to the effectiveness of the drop loss for dealing with the problem of imbalanced noisy-labeled samples. This experiment further validates the good generalization ability of the proposed drop loss for facial attribute recognition.

We also compare the average accuracy obtained by the proposed DropNet and several state-of-the-art facial attribute recognition methods (including PANDA [67], LNets+ANet [12], Adaptive Weight [24], MCNN-AUX [9], and GNAS [25]) on the CelebA and LFWA test datasets, as given in Table III. We can see that the proposed method achieves

TABLE IV
PEDESTRIAN ATTRIBUTE RECOGNITION ON THE MARKET-1501 ATTRIBUTE TEST DATASET. METRIC: BALANCED ACCURACY(%). THE BEST RESULTS ARE BOLDFACED.

Methods	*bal-acc*
APR [2]	73.39
Adaptive Weight [24]	74.30
MOON [8]	78.36
Downsampling [68]	74.53
Oversampling [68]	76.69
Cost-sensitive [52]	76.03
GHM-C [33]	74.41
GCEL [44]	78.58
SCN [47]	69.16
Baseline	71.06
DropNet	**81.13**

TABLE V
PEDESTRIAN ATTRIBUTE RECOGNITION ON THE DUKEMTMC ATTRIBUTE TEST DATASET. METRIC: AVERAGE BALANCED ACCURACY (%). THE BEST RESULTS ARE BOLDFACED.

Methods	*bal-acc*
APR [2]	71.12
Adaptive Weight [24]	71.53
Downsampling [68]	73.58
Oversampling [68]	74.29
Cost-sensitive [52]	72.39
MOON [8]	75.67
GHM-C [33]	71.10
GCEL [44]	76.67
SCN [47]	54.96
Baseline	69.52
DropNet	**77.13**

better or comparable performance than the other competing methods, which demonstrates the superiority of the proposed drop loss. Different from the competing methods that usually design sophisticated CNN architectures, DropNet only relies on the simple ResNet-50 model based on the drop loss.

D. Comparisons on Pedestrian Attribute recognition

In this section, we compare the proposed DropNet with several state-of-the-art methods on the task of pedestrian attribute recognition.

1) Competitors: In addition to the above methods (i.e., Downsampling, Oversampling, Cost-sensitive, MOON, GHM-C, GCEL, and SCN), we also compare DropNet with several state-of-the-art pedestrian attribute recognition methods, including APR [2] and Adaptive Weight [24]. All models use the same training set and test set.

2) Overall Evaluation: Table IV and Table V show the balanced accuracy and average balanced accuracy obtained by all the competing methods on the Market-1501 attribute and DukeMTMC attribute test datasets, respectively. The balanced accuracy obtained by each attribute on the Market-1501 attribute is given in the supplementary material.

The proposed DropNet outperforms all pedestrian attribute recognition methods on two datasets with a large margin, which shows the effectiveness of the proposed method for the task of pedestrian attribute recognition. Specifically, our DropNet achieves much better average balanced accuracy than the state-of-the-art pedestrian attribute recognition methods (i.e., APR and Adaptive Weight) on the Market-1501 attribute and DukeMTMC attribute (about 7.74%, 6.83% and 6.01%, 5.60% improvements, respectively). In particular, compared with the imbalanced learning methods, DropNet still outperforms the best competitor MOON by 2.77% and 1.46% on the Market-1501 attribute and DukeMTMC attribute datasets, respectively. This further verifies the significant superiority and generalization of the drop loss in coping with extremely imbalanced attribute data (note that the attribute imbalance ratios of the Market-1501 attribute and DukeMTMC attribute are up to 1:445 and 1:611, respectively).

TABLE VI
PEDESTRIAN ATTRIBUTE RECOGNITION ON THE MARKET-1501 AND DUKEMTMC ATTRIBUTE TEST DATASETS. METRIC: AVERAGE ACCURACY (%). THE BEST RESULTS ARE BOLDFACED.

Methods	*acc*	
	Market-1501	DukeMTMC
PANDA [67]	86.84	85.91
Ped_attrib_net [2]	86.19	82.39
APR [2]	88.16	86.42
Separate Models [24]	86.68	85.45
Adaptive Weight [24]	88.49	87.53
GNAS [25]	88.83	-
GCEL [44]	93.17	90.78
SCN [47]	89.96	83.99
Baseline	91.89	89.76
DropNet	**93.81**	**91.24**

3) Further Results: We also show the average accuracy obtained by different competing methods (including PANDA [67], Ped_attrib_net [2], APR [2], Separate Models [24], Adaptive Weight [24], and GNAS [25]) on the Market-1501 and DukeMTMC attribute test datasets in Table VI. The proposed DropNet outperforms the other competing methods in terms of average accuracy on two different datasets. This is because the drop loss effectively alleviates the problem of imbalanced noisy-labeled samples for pedestrian attribute recognition.

E. Further Evaluations and Discussions

In this section, we perform extensive experiments to analyze the effectiveness of noisy-labeled sample modeling in DropNet and the influence of the key parameters on the final performance.

1) Effectiveness of Noisy-Labeled Sample Modeling: In this subsection, we evaluate the effectiveness of noisy-labeled sample modeling.

First, following the common settings in [49], [61], we add symmetric noise to the MAAD-Face dataset, where the noise rate of all attributes is gradually increased from 0% to 60%. We report the average accuracy and average balanced accuracy in Table VII. We also show the performance obtained by the baseline method and SCN [47].

In Table VII, DropNet obtains much better performance on both average balanced accuracy and average accuracy than SCN, when the noise rate changes from 0% to 40%. This result demonstrates the superiority of our proposed noisy-labeled sample modeling technique, which selects noisy-labeled samples according to their gradient norms. However, DropNet under the setting of Symmetric-60% achieves much worse performance than that under other noise settings. This is due to the negative influence of a large proportion of noisy labels.

TABLE VII
FACIAL ATTRIBUTE RECOGNITION UNDER DIFFERENT NOISE RATES ON THE MAAD-FACE DATASET. METRIC: AVERAGE ACCURACY (%) AND AVERAGE BALANCED ACCURACY (%).

Methods	0%		Symmetry-20%		Symmetry-40%		Symmetry-60%	
	acc	*bal-acc*	*acc*	*bal-acc*	*acc*	*bal-acc*	*acc*	*bal-acc*
Baseline	84.44	65.94	83.05	62.84	81.39	60.92	62.14	52.32
SCN [47]	78.90	56.96	75.92	55.24	71.79	52.81	62.76	44.18
DropNet	**85.13**	**67.80**	**84.12**	**65.89**	**82.32**	**64.41**	**63.01**	**52.51**

Second, we show the trend of the drop rate vs. the number of epochs and that of the label F1-score vs. the number of epochs for four attributes (including “High_Cheekbones”, “Wearing_Necktie”,“Mouth_Closed”, and “Eyeglasses”) under two different noise settings (including Symmetry-20% and Symmetry-40%) in Fig. 4.

As shown in the upper row of Fig. 4, samples are dropped at an increasing rate in the early stage of the training process and then the number of dropped samples remains stable in the later stage of the training process. This is because we gradually reduce the value of threshold α (which is used to determine the noisy-labeled candidates) by using the linear decrease strategy for the first T epochs of the training stage, thus leading to more dropped samples. In this way, our method is able to learn simple and general patterns from clean samples before fitting noisy-labeled samples. After T epochs, the threshold α is fixed and thus the number of noisy labeled candidates is stable.

As shown in the lower row of Fig. 4, the proposed DropNet effectively identifies most of the noisy-labeled samples for different attributes (the label F1-scores of all four attributes are above 0.90) in the training set under the setting of Symmetry-20%. Meanwhile, we can see that different attributes have different label F1-scores. This indicates an imbalance in the numbers of noisy-labeled samples for different attributes. The label F1-score significantly drops under the setting of Symmetry-40% compared with that of Symmetric-20%. Therefore, how to address PAR with high noise rates needs further study. We also visualize some dropped samples for the “Eyeglasses” attribute during different stages of the training, as shown in Fig. 5. We can see that some dropped samples are the clean samples in the early training stage, while most of the dropped samples are noisy-labeled samples in the later training stage. Therefore, our method can more accurately identify noisy-labeled samples as the training progresses.

Then, we perform a toy experiment to evaluate the performance when only two hard attributes (i.e., “Oval_Face” and “High_Cheekbones”) are used. The results are given in the supplementary material.

2) Influence of Parameter λ: In this subsection, we evaluate parameter λ in Eq. (5), which is used to determine the noisy-labeled candidates, on the final performance. We fix the value of β to 1.00 and change the value of λ from 0.00 to 0.50.

Table VIII shows the influence of the parameter λ under two different evaluation metrics (i.e., average balanced accuracy and average accuracy) on the LFWA and DukeMTMC attribute test datasets.

From Table VIII, we can see that when the value of λ is 0.30, DropNet achieves the best performance in terms of average accuracy and average balanced accuracy. The performance is slightly different, when the values of λ are within the range of [0.25,0.45]. This indicates that the drop loss is not very sensitive to the value of α within a certain range.

On the one hand, the informative samples are treated as noisy-labeled candidates, when the value of λ is too large (above 0.50). On the other hand, when the value of λ is too small (below 0.15), some noisy-labeled samples are used for training. Therefore, when the value of λ is too small or too

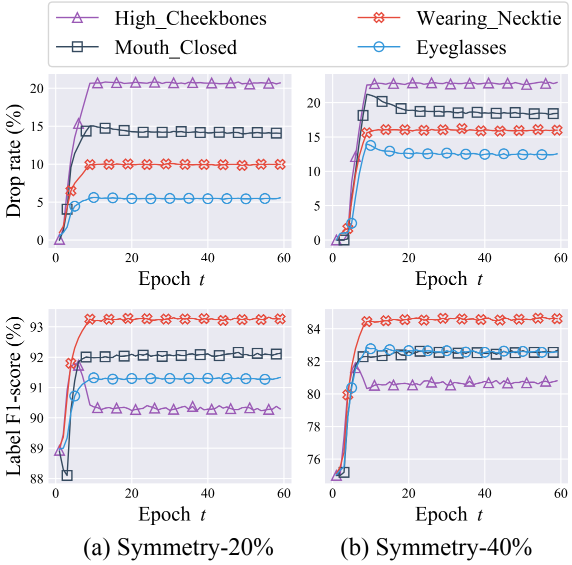

Fig. 4. Ablation studies on the MAAD-Face dataset under the settings of (a) Symmetry-20% noise and (b) Symmetry-40% noise. Four objective attributes (i.e., “High_Cheekbones”, “Wearing_Necktie”,“Mouth_Closed”, and “Eyeglasses”) from the MAAF-Face dataset are employed. Top: drop rate (%) vs. epochs; bottom: label F1-score (%) vs. epochs.

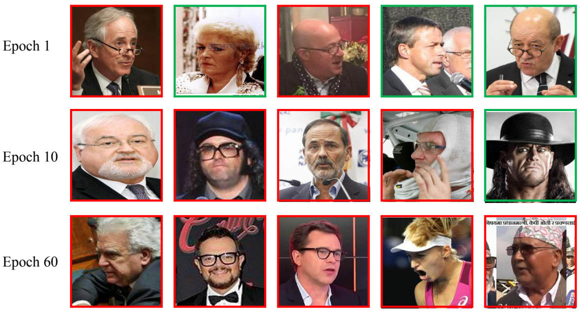

Fig. 5. Dropped samples corresponding to the top-5 largest gradient norms for the “Eyeglasses” attributes in the MAAD-Face dataset in the different training stages under the setting of Symmetry-20% noise. The image with a red border is the noisy-labeled sample, and the green border is the clean sample.

TABLE VIII
PERFORMANCES OF DROPNET WITH DIFFERENT VALUES OF λ ON THE LFWA AND DUKEMTMC ATTRIBUTE TEST DATASETS. METRIC: AVERAGE ACCURACY (%) AND AVERAGE BALANCED ACCURACY (%). THE BEST RESULTS ARE BOLDFACED.

λ	LFWA		DukeMTMC	
	acc	*bal-acc*	*acc*	*bal-acc*
0.50	86.05	82.86	90.05	74.83
0.45	86.10	83.18	90.86	76.41
0.30	**86.52**	**83.25**	**91.24**	**77.13**
0.25	86.10	83.16	90.84	76.65
0.15	86.65	83.10	90.26	75.40
0.00	84.79	78.06	89.76	69.52

large, the performance declines. Especially when the value of λ is set to 0.00, DropNet becomes the CE loss-based model. In this case, the performance of DropNet significantly drops in terms of the average balanced accuracy (i.e., 5.19% and 7.61% decreases on the LFWA attribute and DukeMTMC attribute, respectively).

In all the experiments, we fix the value of λ to 0.30.

3) Influence of Parameter β: In this section, we evaluate the performance of DropNet with different values of β defined in Eq. (8). We fix the value of λ to 0.30 and change the values of β to 0.50, 0.90, and 1.00. We also evaluate the proposed DropNet with fixed drop rates. In particular, DropNet with fixed drop rates refers to the method that drops the noisy-labeled candidates for each attribute using the same drop rate (we manually set the drop rate from 0.80 to 1.00). The comparison results on the LFWA and DukeMTMC attribute test datasets are given in Table IX.

DropNet using adaptive drop rates obtains higher accuracy (in terms of both average balanced accuracy and average accuracy) than that using fixed drop rates. Note that when the drop rate is 1.00, all the noisy-labeled candidates are dropped for each attribute. In this case, DropNet achieves the lowest average balanced accuracy and the lowest average accuracy. The main reason is that the model cannot accurately identify the noisy-labeled samples based on the gradient norms at the initial iterations of the training. As a result, simply dropping the whole noisy-labeled candidate set discards some useful training samples. Furthermore, the imbalanced noisy-labeled samples for multiple attributes are ignored for DropNet with fixed drop rates. DropNet with $\beta = 1.00$ achieves better performance than the other methods. This result demonstrates the importance of our developed adaptive drop rates, where the attributes are treated in an easy-to-hard way.

In all the experiments, we fix the value of β to 1.00.

4) Effectiveness against Different Network Architectures: We report the performance of our proposed DropNet based on different network architectures (including ResNet-18, ResNet-50, and VGG-16) as backbones. The results are shown in Table X. The results obtained by different backbones themselves are also given for a comparison.

Compared with the baseline, DropNet consistently improves the accuracy. This can be ascribed to the effectiveness of the drop loss, which effectively addresses the imbalanced noisy labeled samples. DropNet based on ResNet-50 gives better results than those based on the other two network architectures. Overall, the above experimental results demonstrate the effectiveness of the proposed DropNet with different network architectures.

TABLE IX
PERFORMANCES OF DROPNET WITH DIFFERENT VALUES OF β ON THE LFWA AND DUKEMTMC ATTRIBUTE TEST DATASETS. METRIC: AVERAGE ACCURACY (%) AND AVERAGE BALANCED ACCURACY (%). THE BEST RESULTS ARE BOLDFACED.

Drop rates	LFWA		DukeMTMC	
	acc	*bal-acc*	*acc*	*bal-acc*
0.80	83.34	81.82	89.49	76.03
0.90	83.04	81.76	88.62	75.83
1.00	80.09	80.16	87.13	75.32
DR_j^b ($\beta = 0.50$)	83.67	81.75	88.64	75.94
DR_j^b ($\beta = 0.90$)	84.14	82.03	89.86	75.53
DR_j^b ($\beta = 1.00$)	**86.52**	**83.25**	**91.24**	**77.13**

TABLE X
PERFORMANCES OF DROPNET WITH DIFFERENT NETWORK ARCHITECTURES ON THE LFWA AND DUKEMTMC ATTRIBUTE TEST DATASETS. METRIC: AVERAGE ACCURACY (%) AND AVERAGE BALANCED ACCURACY (%). THE BEST RESULTS ARE BOLDFACED.

Network	LFWA		DukeMTMC	
	acc	*bal-acc*	*acc*	*bal-acc*
ResNet-18	81.32	75.90	85.49	64.34
DropNet (ResNet-18)	83.30	80.01	88.58	71.19
ResNet-50	84.79	78.06	89.76	69.52
DropNet (ResNet-50)	**86.52**	**83.25**	**91.24**	**77.13**
VGG-16	83.21	76.57	89.12	68.33
DropNet (VGG-16)	85.29	82.88	91.03	75.42

V. CONCLUSION

We propose a novel and simple drop loss to effectively deal with the problem of imbalanced noisy-labeled samples for PAR. In the drop loss, the noisy-labeled candidates for different attributes are progressively dropped according to adaptive drop rates. Based on the drop loss, a ResNet-50 model (termed DropNet) is trained. Extensive experiments have shown the effectiveness of the proposed DropNet in comparison to several state-of-the-art PAR methods in terms of both balanced accuracy and classification accuracy on the tasks of facial attribute recognition and pedestrian attribute recognition.

Apart from the problem of imbalanced noisy-labeled samples, data in PAR usually exhibit great class imbalance. How to jointly alleviate these two closely related imbalanced learning problems of PAR needs more investigation. Furthermore, we believe that our proposed drop loss is general and can be applied to other computer vision tasks (such as multi-label image classification) involving multi-attribute learning with noisy-labeled samples. In future work, we intend to explore more applications of drop loss.

REFERENCES

[1] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Describable visual attributes for face verification and image search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 10, pp. 1962–1977, 2011.

[2] Y. Lin *et al.*, "Improving person re-identification by attribute and identity learning," *Pattern Recognit.*, vol. 95, pp. 151–161, 2019.

[3] J. Zheng, Z. Jiang, R. Chellappa, and J. P. Phillips, "Submodular attribute selection for action recognition in video," in *Adv. Neural Inf. Process. Syst.*, 2014, pp. 1341–1349.

[4] Z. Li, J. Liu, J. Tang, and H. Lu, "Robust structured subspace learning for data representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 10, pp. 2085-2098, 2015.

[5] N. Kumar, P. Belhumeur, and S. Nayar, "FaceTracer: A search engine for large collections of images with faces," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 340–353.

[6] L. Bourdev, S. Maji, and J. Malik, "Describing people: A poselet-based approach to attribute classification," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 1543–1550.

[7] N. Thom, E. M. Hand, "Facial attribute recognition: A survey," Comput. Vis.: A Reference Guide, pp. 1–13, 2020.

[8] E. M. Rudd, M. Günther, and T. E. Boult, "MOON: A mixed objective optimization network for the recognition of facial attributes," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 19–35.

[9] E. M. Hand and R. Chellappa, "Attributes for improved attributes: A multi-task network utilizing implicit and explicit relationships for facial attribute classification," in *Proc. AAAI Conf. Art. Intell.*, 2017.

[10] J. Jang, H. Cho, J. Kim, J. Lee, and S. Yang, "Facial attribute recognition by recurrent learning with visual fixation," *IEEE Trans. Cybern.*, vol. 49, no. 2, pp. 616–625, 2019.

[11] X. Lin *et al.*, "Task-oriented feature-fused network with multivariate dataset for joint face analysis," *IEEE Trans. Cybern.*, vol. 50, no. 3, pp. 1292–1305, 2020.

[12] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 3730–3738.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[14] G. Hu *et al.,* "Attribute-enhanced face recognition with neural tensor fusion networks," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 3764–3773.

[15] S. M. Iranmanesh, H. Kazemi, S. Soleymani, A. Dabouei, and N. M. Nasrabadi, "Deep sketch-photo face recognition assisted by facial attributes," in *Proc. IEEE Int. Conf. Biom. Theory Appl. Syst.*, 2018, pp. 1–10.

[16] Z. Li, J. Tang, and T Mei, "Deep collaborative embedding for social image understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2070–2083, 2019.

[17] Z. Peng, Z. Li, J. Zhang, Y. Li, G.-J. Qi, and J. Tang, "Few-shot image recognition with knowledge transfer," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 441–449.

[18] Z. Li, Y. Sun, L. Zhang, and J. Tang, "CTNet: Context-based tandem network for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021.

[19] A. H. Abdulnabi, G. Wang, J. Lu, and K. Jia, "Multi-task CNN model for attribute prediction," *IEEE Trans. Multimedia*, vol. 17, no. 11, pp. 1949–1959, 2015.

[20] H. Han, A. K. Jain, F. Wang, S. Shan, and X. Chen, "Heterogeneous face attribute estimation: A deep multi-task learning approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 11, pp. 2597–2609, 2017.

[21] J. Li, F. Zhao, J. Feng, S. Roy, S. Yan, and T. Sim, "Landmark free face attribute prediction," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4651–4662, 2018.

[22] D. Li, X. Chen, Z. Zhang, and K. Huang, "Pose guided deep model for pedestrian attribute recognition in surveillance scenarios," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2018, pp. 1–6.

[23] Z. Tan, Y. Yang, J. Wan, H. Hang, G. Guo, and S. Z. Li, "Attention-based pedestrian attribute analysis," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 6126–6140, 2019.

[24] K. He, Z. Wang, Y. Fu, R. Feng, Y.-G. Jiang, and X. Xue, "Adaptively weighted multi-task deep network for person attribute classification," in *Proc. ACM Int. Conf. Multimedia*, 2017, pp. 1636–1644.

[25] S. Huang, X. Li, Z.-Q. Cheng, Z. Zhang, and A. Hauptmann, "GNAS: A greedy neural architecture search method for multi-attribute learning," in *Proc. ACM Int. Conf. Multimedia*, 2018, pp. 2049–2057.

[26] Y. Shu, Y. Yan, S. Chen, J.-H. Xue, C. Shen, and H. Wang, "Learning spatial-semantic relationship for facial attribute recognition with limited labeled data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 11916–11925.

[27] B. Lingenfelter and E. M. Hand, "Improving evaluation of facial attribute prediction models," in *Proc. IEEE Conf. Autom. Face Gesture Recognit.*, 2021, pp. 1–7.

[28] J. Jia, C. Chen, and K. Huang, "Spatial and semantic consistency regularizations for pedestrian attribute recognition," in *Proc. Int. Conf. Comput. Vis.*, 2021, pp. 962–971.

[29] S. Aslan, U. Güdükbay, and H. Dibeklioğlu, "Multimodal assessment of apparent personality using feature attention and error consistency constraint," *Image Vis. Comput.*, vol. 110, pp. 104163, 2021.

[30] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *J. Big Data*, vol. 6, no. 1, pp. 1–54, 2019.

[31] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Netw.*, vol. 106, pp. 249–259, 2018.

[32] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.

[33] B. Li, Y. Liu, and X. Wang, "Gradient harmonized single-stage detector," in *Proc. AAAI Conf. Art. Intell.*, 2019, pp. 8577–8584.

[34] H. Wang, Z. Cui, Y. Chen, M. Avidan, A. B. Abdallah, and A. Kronzer, "Predicting hospital readmission via cost-sensitive deep learning," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 15, no. 6, pp. 1968–1978, 2018.

[35] C. Huang, Y. Li, C. Change Loy, and X. Tang, "Learning deep representation for imbalanced classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5375–5384.

[36] Q. Dong, S. Gong, and X. Zhu, "Imbalanced deep learning by minority class incremental rectification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 6, pp. 1367–1381, 2018.

[37] Z. Yu, D. Wang, Z. Zhao, C. L. P. Chen, J. You, H. Wong, and J. Zhang. "Hybrid incremental ensemble learning for noisy real-world data classification," *IEEE Trans. Cybern.*, vol. 49, no. 2, pp. 403–416, 2019.

[38] S. Xia, Y. Zheng, G. Wang, P. He, H. Li, and Z. Chen, "Random space division sampling for label-noisy classification or imbalanced classification," *IEEE Trans. Cybern.*, 2021.

[39] J. Speth and E. M. Hand, "Automated label noise identification for facial attribute recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019, pp. 25–28.

[40] B. Frénay and M. Verleysen, "Classification in the presence of label noise: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 5, pp. 845–869, 2013.

[41] T. G. Dietterich, "An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization," *Mach. Learn.*, vol. 40, no. 2, pp. 139–157, 2000.

[42] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang, "Learning from massive noisy labeled data for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 2691–2699.

[43] J. Goldberger and E. Ben-Reuven, "Training deep neural-networks using a noise adaptation layer," in *Int. Conf. Learn. Representations*, 2017, pp. 1–9.

[44] Z. Zhang and M. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Adv. Neural Inf. Process. Syst.*, 2018, pp. 8778–8788.

[45] G. Ding, Y. Guo, K. Chen, C. Chu, J. Han, and Q. Dai. "DECODE: Deep confidence network for robust image classification," *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 3752–3756, 2019.

[46] J. Zeng, S. Shan, and X. Chen, "Facial expression recognition with inconsistently annotated datasets," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 222–237.

[47] K. Wang, X. Peng, J. Yang, S. Lu, and Y. Qiao, "Suppressing uncertainties for large-scale facial expression recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020.

[48] X. Zhang, X. Zhu, and S. Wright, "Training set debugging using trusted items," in *Proc. AAAI Conf. Art. Intell.*, 2018, pp. 1–10.

[49] J. Huang, L. Qu, R. Jia, and B. Zhao, "O2U-Net: A simple noisy label detection approach for deep neural networks," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 3326–3334.

[50] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. "Curriculum learning," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 41–48.

[51] C. Cortes, V. Vapnik, "Support-vector networks," *Mach. learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[52] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1263–1284, 2009.

[53] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, and R. Togneri, "Cost-sensitive learning of deep feature representations from imbalanced data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 8, pp. 3573–3587, 2017.

[54] C. Zhang, K. C. Tan, H. Li, and G. S. Hong, "A cost-sensitive deep belief network for imbalanced classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 1, pp. 109–122, 2018.

[55] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," *Tech. Rep.* 07-49, Univ. of Massachusetts, 2007.

[56] P. Terhörst, D. Fährmann, J. N. Kolf, N. Damer, F. Kirchbuchner, and A. Kuijper, "MAAD-Face: A massively annotated attribute dataset for face images." *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 3942–3957, 2021.

[57] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman. "VGGFace2: A dataset for recognising faces across pose and age," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit.* , 2018, pp. 67-74.

[58] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 1116–1124.

[59] Z. Zheng, L. Zheng, and Y. Yang, "Unlabeled samples generated by gan improve the person re-identification baseline in vitro," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 3754–3762.

[60] C. Huang, Y. Li, C. L. Chen, and X. Tang, "Deep imbalanced learning for face recognition and attribute prediction," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019.

[61] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," in *Adv. Neural Inf. Process. Syst.*, 2018, pp. 8527–8537.

[62] A. Paszke *et al.*, "Automatic differentiation in PyTorch," in *Adv. Neural Inf. Process. Syst.*, 2017, pp. 1–4.

[63] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[64] D. P. Kingma and J. Ba. (2014). "Adam: A method for stochastic optimization." [Online]. Available: https://arxiv.org/abs/1412.6980

[65] P. Goyal *et al.,* (2017). "Accurate, large minibatch SGD: Training ImageNet in 1 hour." [Online]. Available: https://arxiv.org/abs/1706.02677

[66] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 815–823.

[67] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev, "PANDA: Pose aligned networks for deep attribute modeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1637–1644.

[68] C. Drummond *et al.*, "C4.5, class imbalance, and cost sensitivity: Why under-sampling beats over-sampling," in *Proc. Int. Conf. Mach. Learn. Workshops*, 2003, pp. 1–8.

Supplementary Material of Drop Loss for Person Attribute Recognition with Imbalanced Noisy-Labeled Samples

Yan Yan, *Member, IEEE*, Youze Xu, Jing-Hao Xue, *Senior Member, IEEE*, Yang Lu,
Hanzi Wang, *Senior Member, IEEE*, and Wentao Zhu

Algorithm 1: The training procedure of DropNet.

```
Input: Training data $\mathcal{T}$; The initialized parameters $\Theta$
       of DropNet; The maximal number of iterations
       $I$; The number of batches $B$; The number of
       attributes $M$.
Output: The parameters $\Theta$ of the trained DropNet.
loop = 1;
while $loop \leq I$ do
    $b = 1$;
    while $b \leq B$ do
        Select a batch $\mathcal{I}^b$ from $\mathcal{T}$;
        Obtain the noisy-labeled candidate sets
          $\{\mathcal{O}_j^b\}_{j=1}^M$ via Eq. (4);
        Compute the average gradient norms $\{G_j^b\}_{j=1}^M$
          via Eq. (6);
        Compute the drop rates $\{DR_j^b\}_{j=1}^M$ via Eq. (7);
        Obtain the noisy-labeled sample drop sets
          $\{\mathcal{O}_j^{b*}\}_{j=1}^M$ via Eq. (8);
        Update the batch $\{\mathcal{I}_j^{b*}\}_{j=1}^M$, where
          $\mathcal{I}_j^{b*} = \mathcal{I}^b \setminus \mathcal{O}_j^{b*}$;
        Compute the drop loss $\mathcal{L}_{drop}$ via Eq. (10);
        Update the parameters $\Theta$ according to the
          $\mathcal{L}_{drop}$ by using the stochastic gradient descent;
        $b = b + 1$;
    end
    $loop = loop + 1$;
end
```

I. Training Algorithm

The overall training procedure of DropNet is summarized in Algorithm 1.

II. More Results

First, we give the balanced accuracy obtained by all the competing methods on the Market-1501 attribute test dataset in Table I.

Then, we manually select two hard attributes (i.e., "Oval_Face" and "High_Cheekbones") in the MAAD-Face dataset. We report the performance obtained by our method and the baseline method in Table II, where the noise rates of two attributes are gradually increased from 0% to 40%.

DropNet achieves only a slight decrease in terms of average balanced accuracy, when the noise rates of two attributes change from 0% to 40%. This result demonstrates the effectiveness of our proposed noisy-label sample modeling. However, the performance of the baseline method significantly drops when the noise rates of the two attributes are 40%. Due to the existence of label noise, the CE loss-based CNN model tends to focus on discriminating noisy-labeled samples and ignores truly useful hard samples, leading to the problem of overfitting and a performance decrease. Compared with the baseline method, DropNet achieves better performance when the noise rates range between 0% and 40%.

Y. Yan, Y. Xu, Y. Lu, H. Wang are with the Fujian Key Laboratory of Sensing and Computing for Smart City, School of Informatics, Xiamen University, Xiamen 361005, China (e-mail: xuyouze@stu.xmu.edu.cn; yanyan@xmu.edu.cn; luyang@xmu.edu.cn; hanzi.wang@xmu.edu.cn).

J.-H. Xue is with the Department of Statistical Science, University College London, London WC1E 6BT, UK (e-mail: jinghao.xue@ucl.ac.uk).

W. Zhu is with Zhejiang Lab, Hangzhou 311121, China (e-mail: wentao.zhu@zhejianglab.com).

TABLE I
PEDESTRIAN ATTRIBUTE RECOGNITION ON THE MARKET-1501 ATTRIBUTE TEST DATASET. METRIC: BALANCED ACCURACY (%). THE BEST RESULTS ARE BOLDFACED.

Attributes / Methods	Gender	Black_lower_clothing	Long_lower_clothing	Hair	White_upper_clothing	Backpack	Bag	Teenager	Adult	Gray_lower_clothing	Clothes	Blue_lower_clothing	Black_upper_clothing	Red_upper_clothing	Handbag	
APR [2]	91	**88**	93	83	88	77	59	76	70	68	87	71	88	87	53	
Adaptive Weight [24]	90	87	93	82	89	81	65	75	75	72	82	75	90	92	54	
MOON [8]	90	87	93	84	89	79	65	**78**	77	75	88	77	91	91	57	
Downsampling [68]	91	87	**94**	85	89	83	68	76	72	70	86	71	89	89	56	
Oversampling [68]	91	86	92	84	**90**	81	63	**78**	76	74	84	76	89	91	56	
Cost-sensitive [52]	91	**88**	92	87	**90**	76	63	**78**	76	72	88	77	90	89	56	
GHM-C [33]	90	87	93	83	88	77	64	77	74	73	83	74	87	91	56	
GCEL [44]	**92**	87	**94**	87	88	**84**	**71**	71	72	78	87	79	90	92	**63**	
SCN [47]	78	84	87	77	83	68	57	58	66	65	87	71	80	89	53	
Baseline	86	**88**	91	83	88	70	52	75	63	68	85	64	85	88	50	
DropNet	91	87	94	**88**	**90**	77	68	**78**	**78**	**79**	**89**	**82**	**92**	**93**	62	

Attributes / Methods	Gray_upper_clothing	Brown_lower_clothing	Green_upper_clothing	White_lower_clothing	Blue_upper_clothing	Sleeve_length	Yellow_upper_clothing	Pink_lower_clothing	Purple_upper_clothing	Hat	Green_lower_clothing	Young	Yellow_lower_clothing	Old	Purple_lower_clothing	Average
APR [2]	69	81	81	73	69	58	90	85	86	55	61	61	50	52	**50**	73.39
Adaptive Weight [24]	71	88	82	72	73	54	92	89	89	52	61	53	50	50	**50**	74.30
MOON [8]	75	85	85	79	76	62	92	91	91	70	67	76	71	56	**50**	78.36
Downsampling [68]	70	82	81	75	72	57	90	85	89	62	64	55	50	51	**50**	74.53
Oversampling [68]	75	**88**	78	73	73	61	91	92	90	75	64	**83**	50	54	**50**	76.69
Cost-sensitive [52]	75	78	75	77	76	69	90	92	92	63	58	70	50	52	**50**	76.03
GHM-C [33]	71	81	81	72	69	56	90	88	90	57	63	63	54	51	**50**	74.41
GCEL [44]	**83**	79	**88**	77	**88**	**73**	**93**	86	81	**87**	**84**	50	50	54	**50**	78.58
SCN [47]	57	79	79	62	72	47	81	83	69	90	52	50	50	50	**50**	69.16
Baseline	65	72	71	72	73	56	92	86	92	56	51	60	50	50	**50**	71.06
DropNet	**83**	85	86	**82**	83	**73**	**93**	**94**	**93**	75	65	80	**83**	**61**	**50**	**81.13**

TABLE II
PERFORMANCE COMPARISONS BETWEEN DROPNET AND BASELINE (IN THE BRACKET) WITH 0%–40% NOISE RATES FOR TWO ATTRIBUTES ON THE MAAD-FACE DATASET. METRIC: AVERAGE ACCURACY (%) AND AVERAGE BALANCED ACCURACY (%). OR-O AND OR-H DENOTE THE NOISE RATES OF THE "OVAL_FACE" AND "HIGH_CHEEKBONES" ATTRIBUTES, RESPECTIVELY.

OR-H / OR-O	0%		20%		40%	
	acc	*bal-acc*	*acc*	*bal-acc*	*acc*	*bal-acc*
0%	76.60 (76.46)	74.13 (73.95)	76.03 (75.65)	73.02 (72.50)	75.09 (73.80)	72.61 (69.85)
20%	75.75 (75.25)	73.04 (72.65)	74.53 (73.82)	71.55 (70.75)	73.24 (71.91)	70.63 (68.16)
40%	74.55 (73.29)	72.02 (70.77)	73.50 (71.52)	70.76 (68.46)	72.30 (69.48)	69.15 (65.40)