Imbalanced Deep Learning by Minority Class Incremental Rectification

Qi Dong, Shaogang Gong, and Xiatian Zhu

Abstract—Model learning from class imbalanced training data is a long-standing and significant challenge for machine learning. In particular, existing deep learning methods consider mostly either class balanced data or moderately imbalanced data in model training, and ignore the challenge of learning from significantly imbalanced training data. To address this problem, we formulate a class imbalanced deep learning model based on batch-wise incremental minority (sparsely sampled) class rectification by hard sample mining in majority (frequently sampled) classes during model training. This model is designed to minimise the dominant effect of majority classes by discovering sparsely sampled boundaries of minority classes in an iterative batch-wise learning process. To that end, we introduce a Class Rectification Loss (CRL) function that can be deployed readily in deep network architectures. Extensive experimental evaluations are conducted on three imbalanced person attribute benchmark datasets (CelebA, X-Domain, DeepFashion) and one balanced object category benchmark dataset (CIFAR-100). These experimental results demonstrate the performance advantages and model scalability of the proposed batch-wise incremental minority class rectification model over the existing state-of-the-art models for addressing the problem of imbalanced data learning.

Index Terms—Class imbalanced deep learning, Multi-label learning, Inter-class boundary rectification, Hard sample mining, Facial attribute recognition, Clothing attribute recognition, Person attribute recognition.

1 Introduction

Machine learning from class imbalanced data, in which the distribution of training data across different object classes is significantly skewed, is a long-standing problem [1,2,3]. Most existing learning algorithms produce *inductive bias* (learning bias) towards the frequent (majority) classes if training data are not balanced, resulting in poor minority class recognition performance. However, accurately detecting minority classes is often important, e.g. in rare event discovery [4]. A simple approach to overcoming class imbalance in model learning is to re-sample the training data (a pre-process), e.g. by down-sampling majority classes, over-sampling minority classes, or some combinations [5,6,7]. Another common approach is cost-sensitive learning, which reformulates existing learning algorithms by weighting the minority classes more [8,9,10].

Over the past two decades, a range of class imbalanced learning methods have been developed [13]. However, they mainly investigate the single-label binary-class imbalanced learning problem in small scale data with class imbalance ratios being small, e.g. within 1:100. These methods are limited when applied to learning from big scale data in computer vision. Visual data are often interpreted by multi-label semantics, e.g. web person images with multi-attributes on clothing and facial characteristics. Automatic recognition of these nameable properties is very useful [14,15], but challenging for model learning due to: (1) Very large scale imbalanced training data [10,16,17], with clothing and facial attribute labelled data exhibiting power-law distributions (Fig. 1). (2) Subtle appearance discrepancy between different

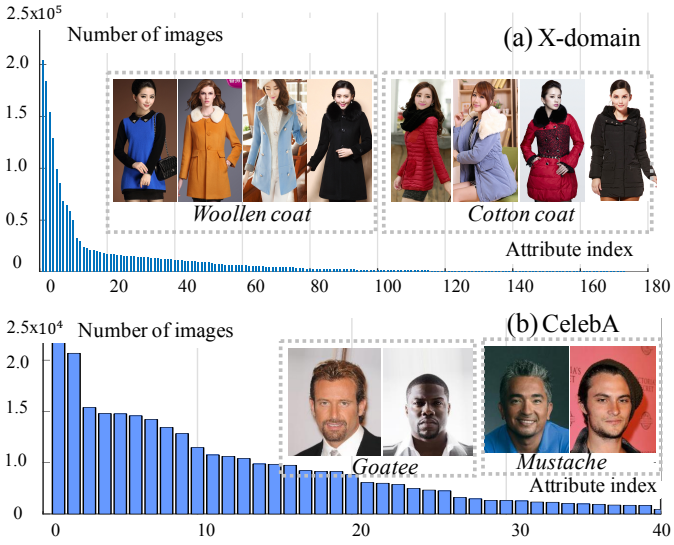


Fig. 1. Imbalanced training data class distributions: (a) clothing attributes (X-Domain [11]), (b) facial attributes (CelebA [12]).

fine-grained attribute classes, e.g. "Woollen-Coat" can appear very similar to "Cotton-Coat", whilst "Mustache" can be visually hard to be distinct (Fig. 1(b)). To discriminate subtle classes from multi-labelled images at large scale, standard learning algorithms require a vast quantity of class *balanced* training data for all labels [11,18].

There is a vast quantity of severely imbalanced visual data on the Internet. Conventional learning algorithms [13, 19] are poorly suited for three reasons: *First*, conventional imbalanced data learning methods without deep learning rely on hand-crafted features extracted from small data, which are inferior to big data deep learning based richer

• *Qi Dong and Shaogang Gong are with the School of Electronic Engineering and Computer Science, Queen Mary University of London, UK. E-mail: {q.dong, s.gong}@qmul.ac.uk. Xiatian Zhu is with Vision Semantics Ltd., London, UK. E-mail: eddy@visionsemantics.com.*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2832629, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 1
Comparing large datasets w.r.t. training data imbalance in terms of class *imbalance ratio* (the sample size ratio between the smallest and largest classes). The ratios are for the standard train/val/test data split if available, otherwise the whole dataset. For MS-COCO [31], no numbers are available for calculating the imbalance ratio, because their images often contain simultaneously multiple classes of objects and multiple instances of a specific class.

ILSVRC2012-14 [32]	MS-COCO [31]	VOC2012 [33]	CIFAR-100 [34]
1 : 2	-	1 : 13	1 : 1
Caltech 256 [35]	CelebA [12]	DeepFashion [30]	X-Domain [11]
1 : 1	1 : 43	1 : 733	**1 : 4,162**

feature representations [20,21,22,23]. *Second*, deep learning in itself also suffers from class imbalanced training data [17,24,25] (Table 9 and Sec. 4.3). *Third*, directly incorporating existing imbalanced data learning algorithms into a deep learning framework does not provide effective solutions [26,27,28].

Overall, imbalanced big data deep learning is understudied partly due to that popular image benchmarks for large scale deep learning, e.g. ILSVRC, do not exhibit significant class imbalance after some careful sample filtering being applied in those benchmark constructions (Table 1). More recently, there are a few emerging large scale clothing and facial attribute datasets that are significantly more imbalanced in class labelled data distributions (Fig. 1), as these datasets are drawn from online Internet sources without artificial sample filtering [11,12,29,30]. For example, the *imbalance-ratio* (lower is more extreme) between the minority classes and the majority classes in the CelebA face attribute dataset [12] is 1:43 (3,713 : 159,057 samples), whilst the X-Domain clothing attributes are even more imbalanced with an imbalance-ratio of 1:4,162 (20 : 204,177) [11] (Table 1).

This work addresses the problem of large scale imbalanced data deep learning for multi-label classification. This problem is characterised by (1) Large scale training data; (2) Multi-label per data sample; (3) Extremely imbalanced training data with an imbalance-ratio being greater than 1:1000; (4) Variable per-label attribute values, ranging from binary to multiple attribute values per label. The **contributions** of this work are: **(I)** We solve the *large scale imbalanced data deep learning* problem. This differs from the conventional imbalanced data learning studies focusing on small scale data single-labelled with a limited number of classes and small data imbalance-ratio. **(II)** We present a novel approach to imbalanced deep learning by minority class incremental rectification using *batch-wise mining* of hard samples on the minority classes in a *batch-wise optimisation* process. This differs from contemporary multi-label learning methods [11,18,29,30,37], which either assume class balanced training data or simply ignore the imbalanced data learning problem all together. **(III)** We formulate a *Class Rectification Loss* (CRL) regularisation algorithm for minority class incremental rectification. In particular, the computational complexity of imposing this rectification loss is restrained by iterative mini-batch-wise model optimisation (small data pools). This is in contrast to the global model optimisation over the entire training data pool of the Large Margin Local Embedding (LMLE) algorithm[1] [17]. There are two advantages of our approach: *First*, the model only requires incremental class imbalanced data learning for all attribute labels concurrently without any additional single-label sampling assumption (e.g. per-label oriented quintuplet construction); *Second*, model learning is independent to the overall training data size, the number of class labels, and without predetermined global data clustering. This makes the model much more scalable to learning from large training data.

Extensive evaluations were performed on the CelebA face attribute [12] and X-Domain clothing attribute [11] benchmarks, with further evaluation on the DeepFashion clothing attribute [30] benchmark. These experimental results show a clear advantage of CRL over 12 state-of-the-art models compared, including 7 attribute models (PANDA [18], ANet [12], Triplet-kNN [39], FashionNet [30], DARN [29], LMLE [17], MTCT [37]). We further evaluated the CRL method on the class balanced single-label object recognition benchmark CIFAR-100 [34], and constructed different class imbalance-ratios therein to quantify the model performance gains under controlled varying degrees of imbalance-ratio in training data.

2 Related Work

Class imbalanced learning aims to mitigate model learning bias towards majority classes by lifting the importance of minority classes [1,2,3]. Existing methods include: (1) ***Data-level***: Aiming to rebalance the class prior distributions in a pre-processing procedure. This scheme is attractive as the only change needed is to the training data rather than to the learning algorithms. Typical methods include down-sampling majority classes, over-sampling minority classes, or both [3,5,6,7,40,41]. However, over-sampling can easily cause model overfitting owing to repeatedly visiting duplicated samples [6]. Down-sampling, on the other hand, throws away valuable information [5,42]. (2) ***Algorithm-level***: Modifying existing algorithms to give more emphasis on the minority classes [8,9,10,43,44,45,46,47,48,49,50]. One strategy is the cost-sensitive learning which assigns varying costs to different classes, e.g. a higher penalty for minority class samples [2,8,9,24,44,50]. However, it is in general difficult to optimise the cost matrix or relationships. Often, it is given by experts therefore problem-specific and non-scalable. In contrast, the threshold-adjustment technique changes the decision threshold in test time [24,51,52,53,54].

1. In LMLE, a computationally expensive data pre-processing (including clustering and quintuplet construction) is required for each round of deep model learning. In particular, to cluster n (e.g. 150,000+) training images w.r.t. an attribute label by k-means, its operation complexity is super-polynomial with the need for at least $2^{\Omega(\sqrt{n})}$ (Ω the lower bound complexity) iterations of cluster refinement on n samples [38]. As each iteration is linear to k and n, a clustering takes the complexity at $k \times O(n) \times 2^{\Omega(\sqrt{n})}$ (O the upper bound complexity). To create a quintuplet for each data sample, four cluster- and class-level searches are needed, each proportion to the training data size n with the overall search complexity as quadratic to n ($O(n^2)$). Given a large scale training set, it is likely that this pairwise search part takes the most significant cost in the pre-processing. Both clustering and quintuplet operations are needed for each attribute label, and their costs are proportional to the total number n_{val} of attribute values, e.g. 80 times for CelebA and 178 times for X-domain. Consequently, the total complexity of the pre-processing per round is $n_{\text{val}} \times \left(k \times O(n) \times 2^{\Omega(\sqrt{n})} + O(n^2)\right)$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2832629, IEEE Transactions on Pattern Analysis and Machine Intelligence

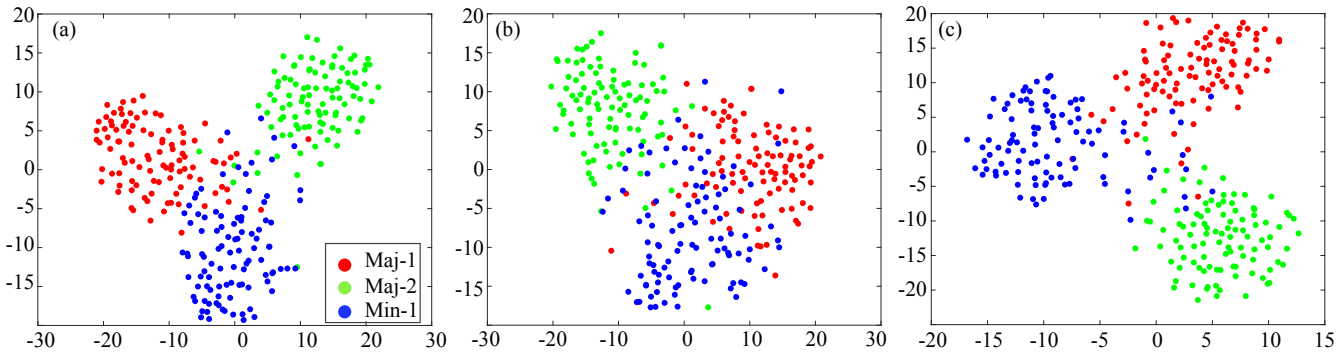


Fig. 2. Visualisation of deep feature distributions learned by the ResNet32 [36] models trained on **(a)** class balanced data, **(b)** class imbalanced data, and **(c)** class imbalanced data with our CRL model. Customised CIFAR-100 object category training image sets were used (more details in Sec. 4.3). In the illustration, we showed only 3 (2 majority and 1 minority) classes for the visual clarity sake. It is observed that the minority class (Min-1) is overwhelmed more severely by the two majority classes (Maj-1, Maj-2) when deploying the existing CNN model given class imbalanced training data. The proposed CRL approach rectifies clearly this undesired model induction bias inherent to existing deep learning architectures.

(3) ***Hybrid***: Combined data-level and algorithm-level rebalancing [55,56,57]. These methods still only consider small scale imbalanced data learning, characterised by: (a) Limited number of data samples and classes, (b) Non-extreme imbalance ratios, (c) Single-label classification, (d) Problem-specific hand-crafted low-dimensional features. In large, these classical techniques are poor for severely imbalanced data learning given big visual data.

There are early neural network based methods for imbalanced data learning [24,25,26,27,28,58,59,60,61]. However, these works still only address small scale imbalanced data learning with neural networks merely acting as nonlinear classifiers without end-to-end learning. A few recent studies [41,62,63,64,65] have exploited classic strategies in single-label deep learning. For example, the binary-class classification problem is studied by per-class mean square error loss [64], synthetic minority sample selection [65], and constant class ratio in mini-batch [66]. The multi-class classification problem is addressed by online cost-sensitive loss [62]. More recently, the idea of preserving local class structures (LMLE) was proposed for imbalanced data deep learning, but without end-to-end model training and with only single-label oriented training unit design [17]. In contrast, our model is designed for end-to-end imbalanced data deep learning for multi-label classification, scalable to large training data.

Hard sample mining has been extensively exploited in computer vision, e.g. object detection [67,68], face recognition [39], image categorisation [69], and unsupervised representation learning [70]. The rational for mining *hard* negatives (unexpected) is that, they are more informative than *easy* negatives (expected) as they violate a model class boundary by being on the wrong side and also far away from it. Therefore, hard negative mining enables a model to improve itself quicker and more effectively with less training data. Similarly, model learning can also benefit from mining hard positives (unexpected), i.e. those on the correct side but very close to or even across a model class boundary. In our model learning, we *only* consider hard mining on the minority classes for efficiency. Moreover, our batch-balancing hard mining strategy differs from that of LMLE [17] by eliminating exhaustive searching of the entire training set (all classes), hence computationally more scalable than LMLE.

Deep metric learning is based on the idea of combining deep neural networks with metric loss functions in a joint end-to-end learning process [39,69,71,72]. Whilst adopting similarly a generic margin based loss function [73,74], deep metric learning does not consider the class imbalanced data learning problem. In contrast, our method is specifically designed to address this problem by incrementally rectifying the structural significance of minority classes in a batch-wise end-to-end learning process, so to achieve scalable imbalanced data deep learning.

Deep learning of clothing and facial attributes has been recently exploited [11,12,18,29,30,37], given the availability of large scale datasets and deep models' strong capacity for learning from big training data. However, existing methods ignore mostly imbalanced class data distributions, resulting in suboptimal model learning and poor model performance on the minority classes. One exception is the LMLE model [17] which studies imbalanced data deep learning [3]. Compared to our end-to-end learning using mini-batch hard sample mining on the minority classes only, LMLE is not end-to-end learning and with global hard mining over the entire training data, it is computationally complex and expensive, not lending itself naturally to big training data.

3 Scalable Imbalanced Deep Learning

For the problem of imbalanced data deep learning from large training data, we consider the problem of person attribute recognition, both facial and clothing attributes. This is a multi-label multi-class learning problem given imbalanced training data, a generalisation of the more common single-label binary-/multi-class recognition problem. Specifically, we wish to construct a deep learning model capable of recognising *multi-labelled* person attributes $\{z_j\}_{j=1}^{n_{\text{attr}}}$ in web images, with a total of n_{attr} different attribute labels. Each *label* z_j has its respective *class* value range Z_j, e.g. multi-class clothing attribute or binary-class facial attribute. Suppose we have n training images $\{\boldsymbol{I}_i\}_{i=1}^{n}$ with their attribute annotation vectors $\{\boldsymbol{a}_i\}_{i=1}^{n}$, and $\boldsymbol{a}_i = [a_{i,1}, \ldots, a_{i,j}, \ldots, a_{i,n_{\text{attr}}}]$ where $a_{i,j}$ refers to the j-th attribute class value of the image $\boldsymbol{I}_i$. The number of images available for different attribute classes varies greatly (Fig. 1) therefore imposing a significant *multi-label imbalanced class data* distribution challenge to model learning. Most

attributes are *localised* to image regions, even though the location information is not annotated (*weakly labelled*). We consider to jointly learn features and *all* the attribute label classifiers from class imbalanced training data in an *end-to-end* process. Specifically, we introduce *incremental minority class discrimination learning* by formulating a Class Rectification Loss (CRL) regularisation. The CRL imposes an additional batch-wise class balancing on top of the cross-entropy loss so to rectify model learning bias due to the over-representation of the majority classes by promoting under-represented minority classes (Fig. 3).

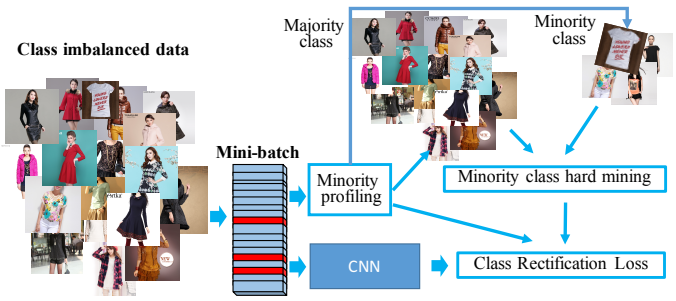


Fig. 3. Overview of the proposed Class Rectification Loss (CRL) regularisation for large scale class imbalanced deep learning.

3.1 Limitations of Cross-Entropy Classification Loss

Convolutional Neural Networks (CNN) are designed to take inputs as two-dimensional images for recognition tasks [75]. For learning a multi-class (per-label) classification CNN model (details in "Network Architecture", Sections 4.1, 4.2, and 4.3), the Cross-Entropy (CE) loss function is commonly used by firstly predicting the j-th attribute posterior probability of $\boldsymbol{I}_i$ over the ground truth $a_{i,j}$:

$$p(y_{i,j} = a_{i,j}|\boldsymbol{x}_{i,j}) = \frac{\exp(\boldsymbol{W}_j^\top \boldsymbol{x}_{i,j})}{\sum_{k=1}^{|Z_j|} \exp(\boldsymbol{W}_k^\top \boldsymbol{x}_{i,j})} \quad (1)$$

where $\boldsymbol{x}_{i,j}$ refers to the feature vector of $\boldsymbol{I}_i$ for the j-th attribute label, and $\boldsymbol{W}_k$ is the corresponding classification function parameter. Then compute the overall loss on a mini-batch of n_{bs} images as the average additive summation of attribute-level loss with equal weight over all labels:

$$\mathcal{L}_{\text{ce}} = -\frac{1}{n_{\text{bs}}} \sum_{i=1}^{n_{\text{bs}}} \sum_{j=1}^{n_{\text{attr}}} \log\Big(p(y_{i,j} = a_{i,j}|\boldsymbol{x}_{i,j})\Big) \quad (2)$$

By design, the cross-entropy loss enforces model learning to respect two conditions: (1) The same-class samples should have class distributions with the identical peak position corresponding to the groundtruth one-hot label. (2) Each class corresponds to a different peak position in the class distribution. As such, the model is supervised end-to-end to separate the class boundaries *explicitly* in the prediction space and *implicitly* in the feature space by some in-between linear or nonlinear transformation. The CE loss minimises the amount of training error by assuming that individual samples and classes are *equally* important. To achieve model generalisation with discriminative inter-class boundary separation, it is necessary to have a large training set with sufficiently *balanced* class distributions (Fig. 2(a)).

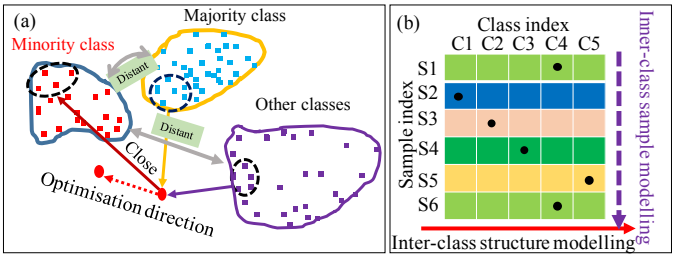


Fig. 4. Illustration of **(a)** inter-class structure rectification around decision boundary in the CRL, as complementary to **(b)** the cross-entropy loss with single-class independent modelling (indicated by dashed arrow).

However, given highly class imbalanced training data, e.g. X-Domain benchmark, model learning by the conventional cross-entropy loss is suboptimal. The model suffers from generalising inductive decision boundaries biased towards majority classes with ignorance on minority classes (Fig. 2(b)). To address this problem, we reformulate the learning objective loss function by *explicitly* imposing structural discrimination of minority classes against others, i.e. *inter-class geometry structure modelling* (Fig. 4). This stresses the structural significance of minority classes in model learning, orthogonal and complementary to the uniform *single-class independent modelling* enforced by the cross-entropy loss (Fig. 4(b)). Conceptually, this design may bring simultaneous benefits to majority class boundary learning as shown in our experimental evaluations (Tables 3 and 5).

3.2 Minority Class Hard Sample Mining

We explore a hard sample mining strategy to enhance minority class manifold rectification by selectively "borrowing" majority class samples from class decision boundary marginal (border) regions. Specifically, we estimate minority class neighbourhood structure by mining *both* hard-positive and hard-negative samples for every selected minority class in *each mini-batch* of training data[2]. Our idea is to rectify *incrementally* the per-batch class distribution bias of multi-labels in model learning. Hence, each improved intermediate model from per-batch training is less inclined towards the over-sampled majority classes and more discriminative to the under-sampled minority classes (Fig. 2(c)). Unlike LMLE [17] which aims to preserve the local structures of *both* majority and minority classes by global clustering of and sampling from the entire training data, our model design aims to enhance progressively minority class discrimination by incremental projective structure refinement. This idea is inherently compatible with *batch-wise* hard-positive and hard-negative sample mining along the model training trajectory. This eliminates the LMLE's drawback in assuming that local group structures of all classes can be estimated reliably by offline *global* clustering before model learning.

Incremental Batch-Wise Class Profiling For hard sample mining, we first profile the minority and majority classes per label in each training mini-batch with n_{bs} training samples. We profile the class distribution $\boldsymbol{h}^j = [h_1^j, \ldots, h_k^j, \ldots h_{|Z_j|}^j]$

2. We *only* consider those minority classes having at least two sample images or more in each batch, i.e. ignoring those minority classes having only one sample image or none. This enables a more flexible loss function selection, e.g. triplet loss functions which typically requires at least two matched samples.

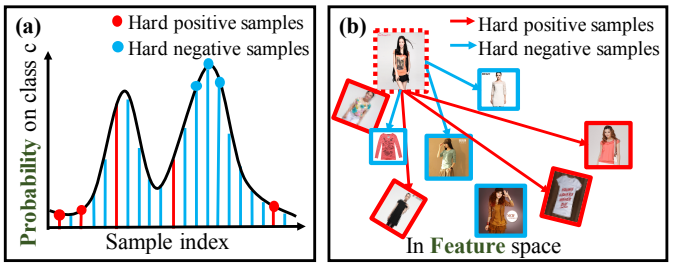


Fig. 5. Illustration of minority class hard sample mining. **(a)** *Class-level mining*: For each minority class c, hard-positives are those samples from class c but with low class prediction scores on c by the current model (red solid circles). Hard-negatives are those with high class c prediction scores but on the wrong class (blue solid circles). **(b)** *Instance-level mining*: For each sample (dotted red box) of a minority class c, hard-positives are the samples of class c (solid red box) further away from the given sample of class c in the feature space (pointed by a red arrow). Hard-negatives are those close to the given sample but from different classes (pointed by blue arrow). Top-3 hard positive and negative samples are shown in the two examples for conceptual illustration.

over Z_j class for each attribute (label) j, where h_k^j denotes the number of training samples with the j-th attribute value assigned to class k. We then sort h_k^j in the descent order. As such, we define the minority classes for attribute label j in this *mini-batch* as the smallest classes C_{min}^j subject to:

$$\sum_{k \in C_{\text{min}}^j} h_k^j \leq \rho \cdot n_{\text{bs}} \tag{3}$$

In the most studied two-class setting [3], the minority (majority) class is defined as the one with fewer (more) samples, i.e. under (above) 50%. However, to our best knowledge there is no standard definition for the multi-class case. For the definition in Eqn. (3) being conceptually consistent to the two-class setting, we also set ρ=50%. This means that *all* minority classes *collectively* account for *at most half or less* samples per batch. The remaining classes are deemed as the majority classes. We analysed the effect of choosing different ρ values on model performance (Table 13).

Given the minority classes, we then consider hard mining therein at two levels: class-level (Fig. 5(a)) and instance-level (Fig. 5(b)). Let us next define the "hardness" metrics, hard samples and their selection.

Hardness Metrics For hard sample mining, it is necessary to have quantitative metrics for "hardness" measurement. Two metrics are considered: (1) *Score based*: A model's class prediction score, suitable for class-level hard mining. (2) *Feature based*: The feature distance between data points, suitable for instance-level hard mining.

Class-Level Hard Samples At the class-level, we quantify the sample hardness regarding a given class per label. Particularly, for any minority class c of the attribute label j, we refer "hard-positives" to the images $\boldsymbol{x}_{i,j}$ of class c ($a_{i,j} = c$ with $a_{i,j}$ the groundtruth class of the attribute j) given *low* prediction scores $p(y_{i,j} = c|\boldsymbol{x}_{i,j})$ on class c by the thus-far model, i.e. *poor* recognitions. Conversely, by "hard-negatives", we refer to the images $\boldsymbol{x}_{i,j}$ of other classes ($a_{i,j} \neq c$) given *high* prediction scores on class c by thus-far model, i.e. *obvious* mistakes. Formally, we define them as:

$$\mathcal{P}_{c,j}^{\text{cls}} = \{\boldsymbol{x}_{i,j} | a_{i,j} = c, \text{ low } p(y_{i,j} = c|\boldsymbol{x}_{i,j})\} \tag{4}$$

$$\mathcal{N}_{c,j}^{\text{cls}} = \{\boldsymbol{x}_{i,j} | a_{i,j} \neq c, \text{ high } p(y_{i,j} = c|\boldsymbol{x}_{i,j})\} \tag{5}$$

where $\mathcal{P}_{c,j}^{\text{cls}}$ and $\mathcal{N}_{c,j}^{\text{cls}}$ denote the hard positive and negative sample sets of a minority class c of the attribute label j.

Instance-Level Hard Samples At the instance-level, we quantify the sample hardness regarding any specific sample instance $\boldsymbol{x}_{i,j}$ (groundtruth class $a_{i,j} = c$) from each minority class c of the attribute label j. Specifically, we define "hard-positives" of $\boldsymbol{x}_{i,j}$ as those class c images $\boldsymbol{x}_{k,j}$ (groundtruth class $a_{k,j} = c$) by thus-far model with *large* distances (low matching scores) from $\boldsymbol{x}_{i,j}$ in the feature space. In contrast, we define "hard-negatives" as those images $\boldsymbol{x}_{k,j}$ not from class c ($a_{k,j} \neq c$) with *small* distances (high matching scores) to $\boldsymbol{x}_{i,j}$ in the feature space. We formally define them as:

$$\mathcal{P}_{i,c,j}^{\text{ins}} = \{\boldsymbol{x}_{k,j} | a_{k,j} = c, \text{ large dist}(\boldsymbol{x}_{i,j}, \boldsymbol{x}_{k,j})\} \tag{6}$$

$$\mathcal{N}_{i,c,j}^{\text{ins}} = \{\boldsymbol{x}_{k,j} | a_{k,j} \neq c, \text{ small dist}(\boldsymbol{x}_{i,j}, \boldsymbol{x}_{k,j})\} \tag{7}$$

where $\mathcal{P}_{i,c,j}^{\text{ins}}$ and $\mathcal{N}_{i,c,j}^{\text{ins}}$ are the hard positive and negative sample sets w.r.t. sample instance $\boldsymbol{x}_{i,j}$ of minority class c in the attribute label j, dist$(\cdot)$ is the Euclidean distance metric.

Hard Mining Intuitively, mining hard-positives enables the model to discover and expand sparsely sampled minority class boundaries, whilst mining hard-negatives aims to efficiently improve the margin structures of minority class boundary corrupted by visually similar *distracting* classes. To facilitate and expedite model training, we adopt the top-κ hard samples mining (selection) strategy. Specifically, at training time, for a minority class c of attribute label j (or a minority class instance $\boldsymbol{x}_{i,j}$) in each training batch data, we select κ hard-positives as the bottom-κ scored on c (or bottom-κ (largest) distances to $\boldsymbol{x}_{i,j}$), and κ hard-negatives as the top-κ scored on c (or top-κ (smallest) distance to $\boldsymbol{x}_{i,j}$), given the current model (or feature space).

Remarks The proposed hard sample mining strategy encourages model learning to concentrate particularly on either *weak* recognitions or *obvious* mistakes when discriminating sparsely sampled class margins of the minority classes. In doing so, the overwhelming bias towards the majority classes in model learning is mitigated by *explicitly* stressing minority class discriminative boundary characteristics. To avoid useful information of unselected "easier" data samples being lost, we perform scalable hard sample mining *independently* in each mini-batch during model training and *incrementally* so over successive mini-batches. As a result, all training samples are utilised randomly in the full learning cycle. Our model can facilitate naturally both class prediction score and instance feature distance based matching. The experiments show that class score rectification yields superior performance due to a better compatibility effect with the score based cross-entropy loss.

3.3 Minority Class Neighbourhood Rectification

We introduce a Class Rectification Loss (CRL) regularisation $\mathcal{L}_{\text{crl}}$ to rectify model learning bias of the standard CE loss (Eqn. (2)) due to class imbalanced training data. This is achieved by incrementally reinforcing the minority class decision boundary margins with CRL aiming to discover latent class boundaries whilst maximising their discriminative margins either directly in the decision score space or indirectly in the feature space. We design the CRL regularisation by the learning-to-rank principle [71,73,74] specifically on

the minority class hard samples, and re-formulate the model learning objective loss function Eqn. (2) as:

$$\mathcal{L}_{\text{bln}} \;=\; \alpha\,\mathcal{L}_{\text{crl}} + (1-\alpha)\,\mathcal{L}_{\text{ce}}, \quad \alpha = \eta\,\Omega_{\text{imb}} \tag{8}$$

where α is a parameter designed to be linearly proportional to a training *class imbalance measure* Ω_{imb}. Given different individual class data sample sizes, we define Ω_{imb} as the minimum percentage count of data samples required over all classes in order to form an overall uniform (i.e. balanced) class distribution in the training data. Eqn. (8) imposes an imbalance-adaptive learning mechanism in CRL regularisation – more weighting is assigned to more imbalanced labels[3], whilst less weighting for less imbalanced labels. Moreover, η is independent of the per-label imbalance, therefore a model hyper-parameter estimated by cross-validation (independent of individual class imbalance). In this study, we explore three loss criteria for $\mathcal{L}_{\text{crl}}$ at *both* class-level and instance-level.

(I) Relative Comparison First, we consider the seminal triplet ranking loss [73] to model the relative relationship constraint between intra-class and inter-class. Considering the small number of training samples in minority classes, it is sensible to make full use of them in order to effectively handle the underlying learning bias. Hence, we regard each minority class sample as an "anchor" in the triplet construction to compute the batch loss balancing regularisation.

Specifically, for each anchor sample $\boldsymbol{x}_{a,j}$, we first construct a set of triplets based on the mined top-κ hard-positives and hard-negatives associated with either the corresponding class c of attribute label j (for class-level hard miming), or the sample instance itself $\boldsymbol{x}_{a,j}$ (for instance-level hard mining). In this way, we form at most κ^2 triplets $T = \{(\boldsymbol{x}_{a,j}, \boldsymbol{x}_{+,j}, \boldsymbol{x}_{-,j})_s\}_{s=1}^{\kappa^2}$ w.r.t. $\boldsymbol{x}_{a,j}$, and a total of at most $|X_{\text{min}}| \times \kappa^2$ triplets T for all the anchors X_{min} across all the minority classes of every attribute label. We then formulate the following triplet ranking loss to impose a CRL class balancing constraint:

$$\mathcal{L}_{\text{crl}} = \frac{\sum_T \max\left(0,\; m_j + \text{d}(\boldsymbol{x}_{a,j}, \boldsymbol{x}_{+,j}) - \text{d}(\boldsymbol{x}_{a,j}, \boldsymbol{x}_{-,j})\right)}{|T|} \tag{9}$$

where m_j denotes the class margin of attribute j and $\text{d}(\cdot)$ is the distance between two samples. We consider both class-level and instance-level model learning rectifications[4].

For *class-level* rectification, we consider the model predictions between matched and unmatched pairs:

$$\text{d}(\boldsymbol{x}_{a,j}, \boldsymbol{x}_{+,j}) = |p_{a,j} - p_{+,j}|, \quad \text{d}(\boldsymbol{x}_{a,j}, \boldsymbol{x}_{-,j}) = p_{a,j} - p_{-,j} \tag{10}$$

where $p_{*,j}$ denotes the model prediction score of $\boldsymbol{x}_{*,j}$ on the target minority class c of attribute label j, with $* \in \{a, +, -\}$. The intuition is that, the matched pair is constrained to have similar prediction scores on the true class (both directions with absolute values), higher than that of any negative sample by a margin m_j in a single direction (without absolute operation). For the triplet ranking, a fixed inter-class margin is often utilised [39] and we set $m_j = 0.5$ for all attribute labels $j \in \{1, \cdots, n_{\text{attr}}\}$. This ensures a correct classification by the maximum a posteriori probability estimation.

For *instance-level* rectification, we consider the sample pairwise distance in the feature space as:

$$\text{d}(\boldsymbol{x}_{a,j}, \boldsymbol{x}_{*,j}) = \|\boldsymbol{f}_{(a,j)} - \boldsymbol{f}_{(*,j)}\|_2, \tag{11}$$

where $\boldsymbol{f}_{(\cdot,j)}$ denotes the attribute j feature vector of the corresponding image sample. We adopt the Euclidean distance. In this case, the m_j (Eqn. (9)) specifies the class margin in the feature space. We apply a geometrically intuitive design: projecting uniformly all the class centres along a unit circle and using the arc length between nearby centres as the class margin. That is, we set the class margin for attribute j as:

$$m_j = \frac{2\pi}{|Z_j|} \tag{12}$$

where $|Z_j|$ is the number of classes.

(II) Absolute Comparison Second, we consider the contrastive loss [74] to enforce absolute pairwise constraints on positive and negative pairs of minority classes. This constraint aims to optimise the boundary of minority classes by incrementally separating the overlapped (confusing) majority class samples in batch-wise optimisation. Specifically, for each sample $\boldsymbol{x}_{a,j}$ in a minority class c of an attribute j, we use the mined hard samples to build positive $P^+ = \{\boldsymbol{x}_{a,j}, \boldsymbol{x}_{+,j}\}$ and negative $P^- = \{\boldsymbol{x}_{a,j}, \boldsymbol{x}_{-,j}\}$ pairs in each training batch. Intuitively, we require the positive pairs to be close whilst the negative pairs to be far away in either model score or sample feature space. Thus, we define the CRL as:

$$\begin{aligned}\mathcal{L}_{\text{crl}} = \frac{1}{2}\Bigg(&\frac{1}{|P^+|}\sum_{P^+}\text{d}(\boldsymbol{x}_{a,j}, \boldsymbol{x}_{+,j})^2 \;+ \\ &\frac{1}{|P^-|}\sum_{P^-}\max\left(m_{\text{ac}} - \text{d}(\boldsymbol{x}_{a,j}, \boldsymbol{x}_{-,j}),\;\; 0\right)^2\Bigg)\end{aligned} \tag{13}$$

where m_{ac} is the between-class margin, which can be set theoretically to an arbitrary positive number [74]. We compute the average loss separately for positive and negative sets to balance their importance even given different sizes.

For *class-level* rectification, we consider the model prediction scores of pairs as defined in Eqn. (10). We set m_{ac}=0.5 to encourage correct prediction. For *instance-level* rectification, we use the Euclidean distance in the feature space (Eqn. (11)) for pairwise comparison. We empirically set m_{ac}=1, which gives satisfactory converging speed and stability in our experiments.

(III) Distribution Comparison Third, we formulate class rectification for minority classes by modelling the *distribution* relationship of positive and negative pairs (built as in "Absolute Comparison"). This distribution based CRL aims to guide model learning by mining minority class decisive regions *non-deterministically*. In spirit of [71], we represent the distribution of positive (P^+) and negative (P^-) pair sets with histograms $H^+ = [h_1^+, \cdots, h_\tau^+]$ and $H^- = [h_1^-, \cdots, h_\tau^-]$ of τ uniformly spaced bins $[b_1, \cdots, b_\tau]$. We compute the positive histogram H^+ as:

$$h_t^+ = \frac{1}{|P^+|}\sum_{(i,j)\in P^+} \varsigma_{i,j,t} \tag{14}$$

3. Multi-label multi-class, e.g. an attribute label has 6∼55 classes.

4. The maximum operation in Eqn. (9) is implemented by a ReLU (rectified linear unit) in TensorFlow.

where

$$\varsigma_{i,j,t} = \begin{cases} \frac{\mathrm{d}(\boldsymbol{x}_{a,j},\boldsymbol{x}_{+,j})-b_{t-1}}{\Delta}, & \text{if } \mathrm{d}(\boldsymbol{x}_{a,j},\boldsymbol{x}_{+,j}) \in [b_{t-1}, b_t] \\ \frac{b_{t+1}-\mathrm{d}(\boldsymbol{x}_{a,j},\boldsymbol{x}_{+,j})}{\Delta}, & \text{if } \mathrm{d}(\boldsymbol{x}_{a,j},\boldsymbol{x}_{+,j}) \in [b_t, b_{t+1}] \\ 0. & \text{otherwise} \end{cases} \quad (15)$$

and Δ defines the pace length between two adjacent bins. The negative histogram H^- can be constructed similarly. To make minority classes distinguishable from majority classes, we enforce the two histogram distributions as disjoint as possible. Formally, we define the CRL regularisation by how much overlapping between the two distributions:

$$\mathcal{L}_{\text{crl}} = \sum_{t=1}^{\tau} \left(h_t^+ \sum_{k=1}^{t} h_k^- \right) \quad (16)$$

Statistically, this CRL distribution loss estimates the probability that the distance of a random negative pair is smaller than that of a random positive pair, either in the score space or the feature space. Similarly, we consider both *class-level* (Eqn. (10)) and *instance-level* (Eqn. (11)) rectification.

In our experiments (Sec. 4), we compared all six CRL loss designs. By default we deploy the class-level Relative Comparison CRL in our experiments if not stated otherwise.

Further Remarks We do not consider exemplars as anchors from majority classes in CRL because the conventional CE loss can already model the majority classes well given their frequent sampling. As demonstrated in our experiments, additional rectification on majority classes gives some benefit but focusing *only* on minority classes makes the CRL model more cost-effective (Table 12). Due to the batch-wise design, the class balancing effect by our proposed regularisor is incorporated throughout the whole training process progressively. Conceptually, our CRL shares a similar principle to Batch Normalisation [76] in achieving learning scalability.

4 Experiments

Datasets For evaluations, we used both imbalanced and balanced benchmark datasets. Given Table 1, we selected the CelebA [12], X-Domain [11], and CIFAR-100 [34] (see Table 2 for statistics) due to: **(1)** The CelebA provides a *class imbalanced learning* test on multiple *binary-class* facial attributes with imbalance ratios up to 1:43. Specifically, it has 202,599 web in-the-wild images from 10,177 person identities with on average 20 images per person. Each image is annotated by 40 attribute labels. Following [12,17], we used 162,770 images for model training (including 10,000 images for validation), and the remaining 19,867 for test. **(2)** The X-Domain offers an *extremely class imbalanced learning* test on multiple *multi-class* clothing attributes with the imbalance ratios upto 1:4,162. This dataset consists of 245,467 shop images extracted from online retailers. Each image is annotated by 9 attribute labels. Each attribute has a different set of mutually exclusive class values, sized from 6 ("sleeve-length") to 55 ("colour"). In total, there are 178 distinctive attribute classes over the 9 labels. We randomly selected 165,467 images for training (including 10,000 images for validation) and the remaining 80,000 for test. **(3)** The CIFAR-100 provides a *single-label class balanced learning* test. This benchmark contains 100 classes with each having 600 images. This test provides a complementary evaluation of the proposed method against a variety of benchmarking methods, and moreover, facilitates extra in-depth model analysis under simulated class imbalanced settings. We used the standard 490/10/100 training/validation/test split per class [34].

Performance Metrics The classification accuracy [12,37] that treats all classes uniformly is not appropriate for class imbalanced test, as a naive classifier that predicts every test sample as majority classes can still achieve a high overall accuracy although it fails all minority class samples. Since we consider the multi-class imbalanced classification test, the common true/false (positive/negative) rates for binary-class classification are no longer valid. In this work, we adopt the *sensitivity* measure that leads to a *class-balanced* accuracy by considering particularly the class distribution statistics [60] and generalises the conventional binary-class criterion [17]. Formally, we compute the per-class sensitivity based on the classification confusion matrix as:

$$S_i = \frac{n_{(i,i)}}{n_i}, \quad n_i = \sum_{j=1}^{c} n_{(i,j)}, \quad i \in \{1, 2, \cdots, c\} \quad (17)$$

where $n_{(i,j)}$ is the number of class i test samples predicted by a model as class j, and n_i is the size of class i (totally c classes). Therefore, the confusion matrix diagonal refers to correctly classified sample numbers of individual classes whilst the off-diagonal to the incorrect numbers. We define the *class-balanced accuracy* (i.e. mean sensitivity) as:

$$A_{\text{bln}} = \frac{1}{c} \sum_{i=1}^{c} S_i \quad (18)$$

The above metric is for the single-label case. For the multi-label test, we average the mean sensitivity measures over all labels (attributes) to give the overall class-balanced accuracy.

Imbalanced Learning Methods for Comparison We considered five existing class imbalanced learning methods: **(1)** Over-Sampling [5]: A multi-label re-sampling strategy to build a more balanced set before model learning through over-sampling minority classes by random replication. **(2)** Down-Sampling [5]: Another training data re-sampling method based on under-sampling majority classes with random sample removal. **(3)** Cost-Sensitive [3]: A class-weighting strategy by assigning greater misclassification penalties to minority classes and smaller penalties to majority classes in loss design. We assign the class weight as $w_i = \exp(-r_i)$ where r_i specifies the ratio of class i in training data. **(4)** Threshold-Adjustment [54]: Adjusting the model decision threshold in test time by incorporating the class probability prior r_i, e.g. moderating the original model prediction p_i to $\tilde{p}_i = p_i * \exp(-r_i)^T$ where $T \in \{1, 2, 3, 4, 5\}$ is a temperature (softening) parameter estimated by cross validation. Given $\tilde{p}_i$, we then use the maximum a posteriori probability for class prediction. **(5)** LMLE [17]: A state-of-the-art class imbalanced deep learning model exploiting the class structure for improving minority class modelling. For fair comparisons, all the methods were implemented on the same network architecture (details below), with the parameters set by following the authors' suggestions if available or cross-validation. All models were trained on the same training data, and evaluated on the same test data. We adopted the class-level relative comparison CRL for all remaining experiments if not stated otherwise.

TABLE 2
Statistics of the three datasets utilised in our evaluations.

Dataset	Semantics	Labels	Classes	Total Images	Training Images	Test Images
CelebA [12]	Facial Attribute	Multiple (40)	Binary (2)	202,599	162,770 (3,713∼135,779/class)	19,867 (432∼17,041/class)
X-Domain [11]	Clothing Attribute	Multiple (9)	Multiple (6∼55)	245,467	165,467 (13∼132,870/class)	80,000 (4∼64,261/class)
CIFAR-100 [34]	Object Category	Single (1)	Multiple (100)	60,000	50,000 (500/class)	10,000 (100/class)

TABLE 3
Facial attribute recognition on the CelebA benchmark [12]. "*": Class imbalanced learning models. Metric: *Class-balanced accuracy* (%). Instance level hard mining. The 1st/2nd best results are indicated in red/blue. MthOpen: Mouth Open; HighChb: High Cheekbones; HvMkup: Heavy Makeup; WvHair: Wavy Hair; OvFace: Oval Face; PntNose: Pointy Nose; ArEyeb: Arched Eyebrows; BlkHair: Black Hair; StrHair: Straight Hair; BrwHair: Brown Hair; BldHair: Blond Hair; GrHair: Gray Hair; NrwEye: Narrow Eyes; RcdHl: Receding Hairline; 5Shdw: 5 oclock Shadow; BshEb: Bushy Eyebrows; RsChk: Rosy Cheeks; DbChn: Double Chin; EyeGls: Eyeglasses; SdBurn: Sideburns; Mstch: Mustache; PlSkin: Pale Skin.

Attributes / Methods	Attractive	MthOpen	Smiling	Lipstick	HighChb	Male	HvMkup	WvHair	OvFace	PntNose	ArEyeb	BlkHair	Big Lips	Big Nose	Young	StrHair	BrwHair	EyeBag	Earrings	NoBeard
Imbalance ratio (1:x)	1	1	1	1	1	1	2	2	3	3	3	3	3	3	4	4	4	4	4	5
Triplet-kNN [39]	83	92	92	91	86	91	88	77	61	61	73	82	55	68	75	63	76	63	69	82
PANDA [18]	85	93	98	97	89	99	95	78	66	67	77	84	56	72	78	66	85	67	77	87
ANet [12]	87	96	97	95	89	99	96	81	67	69	76	90	57	78	84	69	83	70	83	93
DeepID2 [77]	78	89	89	92	84	94	88	73	63	66	77	83	62	73	76	65	79	74	75	88
Over-Sampling* [5]	77	89	90	92	84	95	87	70	63	67	79	84	61	73	75	66	82	73	76	88
Down-Sampling* [5]	78	87	90	91	80	90	89	70	58	63	70	80	61	76	80	61	76	71	70	88
Cost-Sensitive* [3]	78	89	90	91	85	93	89	75	64	65	78	85	61	74	75	67	84	74	76	88
Threshold-Adjustment* [54]	69	89	88	89	83	95	89	77	72	72	76	86	66	76	24	73	81	76	76	15
LMLE* [17]	88	96	99	99	92	99	98	83	68	72	79	92	60	80	87	73	87	73	83	96
CRL*	81	94	92	95	87	98	90	79	66	71	80	88	67	77	83	72	84	79	84	93

Attributes / Methods	Bangs	BldHair	BshEb	Necklace	NrwEye	5Shdw	RcdHl	Necktie	EyeGls	RsChk	Goatee	Chubby	SdBurn	Blurry	Hat	DbChn	PlSkin	GrHair	Mstch	Bald	**Mean**
Imbalance ratio (1:x)	6	6	6	7	8	8	11	13	14	14	15	16	17	18	19	20	22	23	24	43	
Triplet-kNN [39]	81	81	68	50	47	66	60	73	82	64	73	64	71	43	84	60	63	72	57	75	**72**
PANDA [18]	92	91	74	51	51	76	67	85	88	68	84	65	81	50	90	64	69	79	63	74	**77**
ANet [12]	90	90	82	59	57	81	70	79	95	76	86	70	79	56	90	68	77	85	61	73	**80**
DeepID2 [77]	91	90	78	70	64	85	81	83	92	86	90	81	89	74	90	83	81	90	88	93	**81**
Over-Sampling* [5]	90	90	80	71	65	85	82	79	91	90	89	83	90	76	89	84	82	90	90	92	**82**
Down-Sampling* [5]	88	85	75	66	61	82	79	80	85	82	85	78	80	68	90	80	78	88	60	79	**78**
Cost-Sensitive* [3]	90	89	79	71	65	84	81	82	91	92	86	82	90	76	90	84	80	90	88	93	**82**
Threshold-Adjustment* [54]	93	92	84	62	71	82	83	76	95	82	89	81	89	78	95	83	85	91	86	93	**79**
LMLE* [17]	98	99	82	59	59	82	76	90	98	78	95	79	88	59	99	74	80	91	73	90	**84**
CRL* (Ours)	95	95	84	73	73	89	88	87	99	90	95	87	95	86	99	89	92	96	93	99	**87**

4.1 Comparisons on Facial Attributes Recognition

Competitors We compared the proposed CRL model with 9 existing methods including the 5 class imbalanced learning models above and other 4 state-of-the-art deep learning models for facial attribute recognition on the CelebA benchmark: (1) PANDA [18], (2) ANet [12], (3) Triplet-kNN [39], and (4) DeepID2 [77].

Network Architecture We adopted the 5-layers CNN architecture DeepID2 [77] as the base network for training all class imbalanced learning methods including CRL and LMLE. Training DeepID2 was based on the conventional CE loss (Eqn. (2)). This provides a baseline for evaluations with and without CRL. Moreover, the CRL allows multi-task learning in the spirit of [78,79], with an additional 64-dim FC_2 feature layer and a 2-dim binary prediction layer for each face attribute.

Parameter Settings We trained the CRL from scratch by the learning rate at 0.001, the decay at 0.0005, the momentum at 0.9, the batch size at 256, and the epoch at 921. We set the loss weight η (Eqn. (8)) to 0.01.

Overall Evaluation Facial attribute recognition performance comparisons are shown in Table 3. It is evident that CRL outperforms all competitors including the attribute recognition models and class imbalanced learning methods on the overall mean accuracy. Compared to the best non-imbalanced learning model DeepID2, CRL improves the average accuracy by 6%. Compared to the state-of-the-art imbalanced learning model LMLE, CRL is better on average accuracy by 3%. Other classic imbalanced learning methods perform inferiorly than both CRL and LMLE. In particular, Over-Sampling brings only marginal gain with no clear difference across all imbalance degrees, suggesting that replication based data rebalance is limited in introducing useful information. Cost-Sensitive is largely similar to Over-Sampling. The performance drop by Down-Sampling and Threshold-Adjustment is due to discarding useful data in balancing the class data distribution and imposing potentially inconsistent adjustment to model prediction. This shows that (1) not all class imbalanced learning methods are helpful, and (2) the clear superiority of our batch-wise incremental minority class rectification method in handling biased model learning over alternative methods.

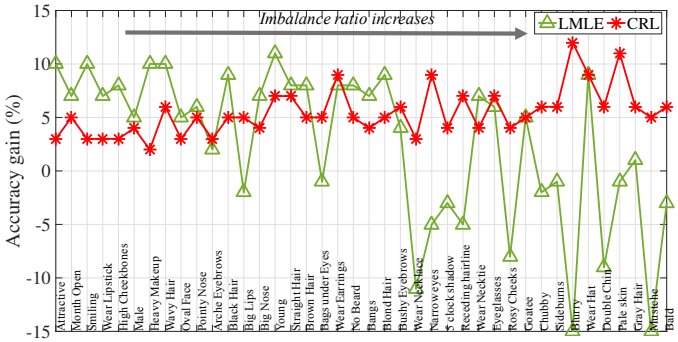


Fig. 6. Performance *additional* gain over the DeepID2 by the LMLE and CRL models on the 40 CelebA binary-class facial attributes [12]. Attributes are sorted from left to right in increasing class imbalance ratio.

TABLE 4
Performance *additional* gain over the DeepID2 by LMLE and CRL on bottom-20 and top-20 CelebA facial attributes in mean *class-balanced accuracy* (%). Negative number means performance drop.

Imbalance Ratio	Bottom-20 (1:1∼1:5)	Top-20 (1:6∼1:43)
LMLE [17]	**+7**	-2
CRL	+5	**+6**

Fig. 7. Examples (3 pairs) of facial attribute recognition (imbalance ratio in bracket). For each pair (attribute), DeepID2 missed both, whilst CRL identified the image with green box but failed the image with red box.

Further Analysis We examined the characteristics of model performance on individual attributes exhibiting different class imbalance ratios. In particular, we further analysed CRL and the best competitor LMLE against the base model DeepID2 without class imbalanced learning. To that end, we split the 40 facial attributes into two groups at a 1:5 imbalance ratio: bottom-20 (the first 20 in Table 3) and top-20 (the remaining) imbalanced attributes. Figure 6 and Table 4 show that: **(1)** CRL improves the prediction accuracy on all attributes (above "0"), whilst LMLE can give weaker prediction than DeepID2 especially on highly imbalanced attributes. This suggests that CRL is more robust in coping with different imbalanced attributes, especially more extremely imbalanced classes. **(2)** LMLE is better at the bottom-20 imbalanced attributes, improving the mean accuracy by 7% *versus* 5% by CRL. For instance, CRL is outperformed by LMLE on the "Attractive" (balanced) and "Heavy Makeup" attributes by 7% and 8%, respectively. This suggests that LMLE is better for less-extremely imbalanced attributes. **(3)** LMLE performance degrades on top-20 imbalanced attributes by 2% in mean accuracy. Specifically, LMLE performs worse than DeepID2 on most attributes with imbalance ratio greater than 1:7, starting from "Wear Necklace" in Table 3. This is in contrast to CRL which achieves an even better performance gain at 6% on top-20. On some very imbalanced attributes, CRL outperforms LMLE significantly, e.g. by 20% on "Mustache" and 27% on "Blurry". Interestingly, the "Blurry" attribute is visually challenging due to its global characteristics not defined by local features therefore very subtle, similar to the "Mustache" attribute (see Fig. 7). This demonstrates that CRL is superior and more scalable than LMLE in coping with severely imbalanced data learning. This is due to (1) incremental batch-wise minority class predictive boundary rectification which is independent to global training data class structure, and (2) end-to-end deep learning for joint feature and classifier optimisation which LMLE lacks.

Model Training Cost Analysis We analysed the model training cost of CRL and LMLE on a workstation with 1 NVIDIA Tesla K40 GPU and 20 E5-2680 @ 2.70GHz CPUs. For LMLE, we used the codes released by the authors[5] with the original settings (4 rounds of training each with $5,000$ iterations of CNN optimisation). The training was initialised by pre-trained DeepID2 face recognition features. On our workstation, LMLE took a total of 264.8 hours to train[6], with each round taking 66.2 hours including 24.5 hours for "clustering+quintuplet construction" and 41.7 hours for "CNN model optimisation". In contrast, CRL took 27.2 hours, that is 9.7 (264.8/27.2) times faster than LMLE.

We further examined model convergence rate. Specifically, LMLE converges quicker than CRL on training batch iterations, LMLE's 20,000 *versus* CRL's 540,000. This is reasonable as LMLE benefits uniquely from both a specifically designed data structural pre-processing (building quintuplets) of the entire training data which is a computationally expensive procedure, and a model pre-training process on auxiliary face recognition labels. However, LMLE is significantly slower than CRL in the overall CNN training time: LMLE's 166.6 hours *versus* CRL's 27.2 hours.

4.2 Comparisons on Clothing Attributes Recognition

Competitors Except the five imbalanced learning methods, we also compared CRL against four other state-of-the-art clothing attribute recognition models: (1) DDAN [11], (2) DARN [29], (3) FashionNet[7] [30], and (4) MTCT [37].

Network Architecture We used the same network structure as the MTCT [37]. Specifically, this network is composited of five stacked NIN conv units [81] and n_{attr} parallel branches with each a 3-FC-layers sub-network for modelling a distinct attribute respectively, in the multi-task learning spirit [79]. We trained MTCT using the CE loss (Eqn. (2)).

Parameter Settings We pre-trained the base network on ImageNet-1K [32] at the learning rate 0.01, then fine-tuned the CRL model on the X-Domain images at a lower learning rate 0.001. We set the decay to 0.0005, the momentum to 0.9, the batch size to 128, and the epoch to 256. We set the loss weight η (Eqn. (8)) to 0.01.

Overall Evaluation Table 5 shows the comparative evaluation of 10 different models on the X-Domain benchmark. It is evident that CRL surpasses all prior state-of-the-art models on all attribute labels. This shows the superiority and scalability of our incremental minority class rectification in tackling extremely imbalanced attribute data, with the maximum imbalance ratio 4,162 *versus* 43 in CelebA attributes. For example, CRL surpasses the best competitor LMLE by 4.65% in mean accuracy. Traditional class imbalanced learning methods behave similarly as on facial attributes, except that Threshold-Adjustment also yields a small gain similar as Cost-Sensitive. Other models without an explicit imbalanced learning mechanism like DDAN, FashionNet, DARN and MTCT suffer notably.

Further Analysis We further examined the performance of CRL and LMLE in comparison to the base model MTCT. Similar to CelebA, we split the 9 attributes into two groups at a 1:5 class imbalance ratio: bottom-1 (the first column

5. The k-means clustering function is not included in the original codes. We used the VLFeat's implementation [80] with the default setting as $1,000$ maximum iterations and 10 repetitions.

6. We did not consider the time cost for pre-training the DeepID2 (needed for extracting the initial features for the first round of data pre-processing) on face identity labels from CelebFaces+ [77] due to lacking the corresponding codes and details. We used the pre-trained DeepID2 model thanks to the helpful sharing by the LMLE authors.

7. We implemented this FashionNet without the landmark detection branch since no landmark labels are available in the X-Domain dataset.

TABLE 5
Clothing attributes recognition on the X-Domain dataset [11]. "*": Imbalanced data learning models. Metric: *Class-balanced accuracy* (%). Slv-Shp: Sleeve-Shape; Slv-Len: Sleeve-Length. The 1st/2nd best results are highlighted in red/blue.

Methods \ Attributes	Category	Colour	Collar	Button	Pattern	Shape	Length	Slv-Shp	Slv-Len	Mean
Imbalance ratio (1:x)	2	138	210	242	476	2,138	3,401	4,115	4,162	
DDAN [11]	46.12	31.28	22.44	40.21	29.54	23.21	32.22	19.53	40.21	**31.64**
FashionNet [30]	48.45	36.82	25.27	43.85	31.60	27.37	38.56	20.53	45.16	**35.29**
DARN [29]	65.63	44.20	31.79	58.30	44.98	28.57	45.10	18.88	51.74	**43.24**
MTCT [37]	72.51	74.68	70.54	76.28	76.34	68.84	77.89	67.45	77.21	**73.53**
Over-Sampling* [5]	73.34	75.12	**71.66**	77.35	77.52	68.98	78.66	67.90	78.19	**74.30**
Down-Sampling* [5]	49.21	33.19	19.67	33.11	22.22	30.33	23.27	12.49	13.10	**26.29**
Cost-Sensitive* [3]	**76.07**	**77.71**	71.24	**79.19**	77.37	69.08	78.08	67.53	77.17	**74.49**
Threshold-Adjustment* [54]	72.51	75.14	71.34	77.91	77.46	70.25	78.78	**70.78**	78.37	**74.72**
LMLE* [17]	75.90	77.62	70.84	78.67	**77.83**	**71.27**	**79.14**	69.83	**80.83**	**75.77**
CRL* (Ours)	**77.69**	**82.01**	**77.01**	**82.37**	**81.39**	**74.96**	**84.81**	**80.48**	**83.02**	**80.42**

TABLE 6
Performance *additional* gain over the MTCT by LMLE and CRL on bottom-1 and top-8 X-Domain clothing attributes in mean accuracy (%).

Imbalance Ratio	Bottom-1 (1:2)	Top-8 (1:138∼1:4,162)
LMLE [17]	+3.39	+2.10
CRL	**+5.18**	**+7.10**

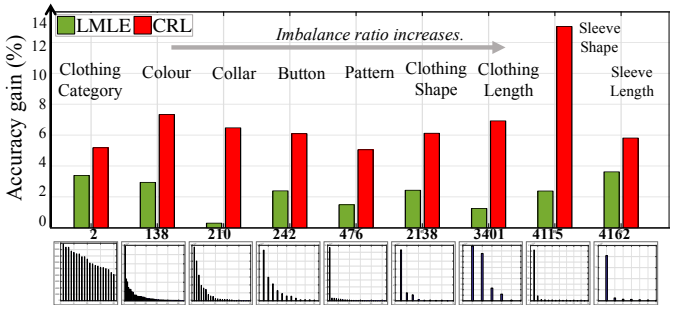


Fig. 8. Performance *additional* gain over the MTCT by the LMLE and CRL models on 9 X-Domain multi-class clothing attributes [11] with the imbalance ratios (numbers under the bars) increasing from left to right.

in Table 5) and top-8 (the remaining). Figure 8 and Table 6 show that: **(1)** LMLE improves MTCT on all clothing attributes with varying imbalance ratios. This suggests that LMLE does address the imbalanced data learning problem in a multi-class setting by embedding local class structures into deep feature learning. **(2)** Compared to LMLE, CRL achieves more significant performance gains on more severely imbalanced attributes. On the top-8 imbalanced attributes, CRL achieves mean accuracy gain of 7.10% *versus* 2.10% by LMLE (Table 6). In particular, our CRL improves LMLE by 10.03% in accuracy for recognising "Sleeve Shape", a fine-grained and visually ambiguous attribute due to its locality and subtle inter-class discrepancy (Fig. 9). This evidence is interesting as it shows that class training data distribution affects a model's ability to learn effectively fine-grained class discrimination. Importantly, a model's ability in coping effectively with class imbalanced data learning can help improve its learning of fine-grained class discrimination. This further demonstrates the strength of CRL over existing models for mitigating model learning bias given severely imbalanced fine-grain labelled classes in an end-to-end deep learning framework.

Model Training Cost Analysis We examined the model training cost of LMLE and CRL on X-Domain using the same workstation as on CelebA. We used the original author released codes with the suggested optimisation setting, e.g. trained the LMLE for 4 rounds each with 5,000 CNN

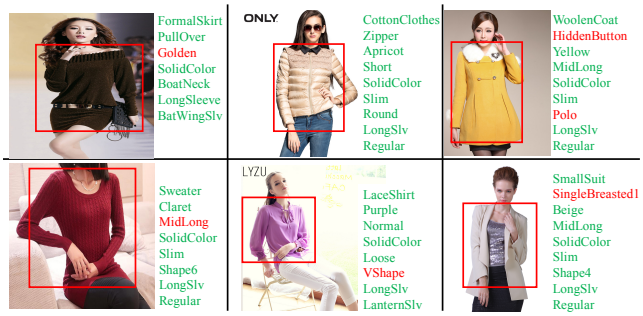


Fig. 9. Examples of clothing attribute recognition by the CRL model, with false attribute prediction in red (Red box: clothing auto-detection).

training iterations. We started with the ImageNet-1K trained VGGNet16 features [20]. For model training, LMLE took 429.9 hours, with each round taking 107.5 hours including 27.6 hours for "clustering+quintuplet construction" and 79.9 hours for "CNN model optimisation". In contrast, CRL took 60.4 hours, that is 7.1 (429.9/60.4) times faster than LMLE.

TABLE 7
Evaluation of CRL on clothing attribute recognition with the DeepFashion benchmark [30]. Metric: *Class-balanced accuracy* (%).

CRL	Texture	Fabric	Shape	Part	Style	*Mean*
ImbRatio (1:x)	733	393	314	350	149	
✗	53.29	52.86	53.02	51.25	51.20	52.20
✓	**55.37**	**55.02**	**55.22**	**53.90**	**53.75**	**54.56**

Further Evaluation We further evaluated the CRL on the DeepFashion clothing attribute dataset [30] with a controlled experiment. We adopted a test setting that is consistent with all the other experiments: (1) The standard multi-label classification setting *without* using the clothing landmark and category labels (used in [30]). (2) ResNet50 [36] as the base network trained by the CE loss. (3) Top-5 attribute predictions in a class-balanced accuracy metric other than a class-biased metric as in [30]. We adopted the standard data split: 209,222/40,000/40,000 images for model training/validation/test. We trained the deep models from scratch with the learning rate as 0.01, the decay as 0.00004, the batch size as 64, and the epoch as 141. We focused on evaluating the additional effect of CRL on top of the CE loss. Table 7 shows that CRL yields a 2.36% (54.56-52.20) boost in mean accuracy.

4.3 Comparisons on Object Category Recognition

We evaluated the CRL on a popular *class balanced* single-label object category benchmark CIFAR-100 [34].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2832629, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 8
Object classification performance (%) on CIFAR-100 [34].

CifarNet	56.5	ResNet32	68.1	DenseNet	74.0
+CRL	**+3.6**	**+CRL**	**+1.2**	**+CRL**	**+0.8**

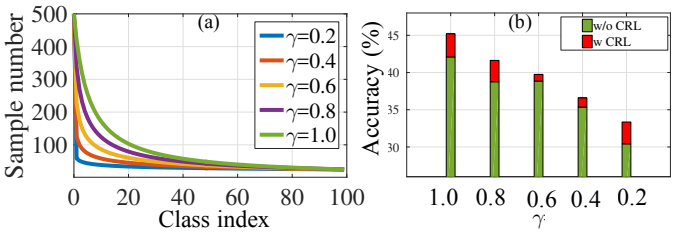


Fig. 10. (a) Simulated imbalanced training data distributions on CIFAR-100 [34]. (b) Performance gains of ResNet32 by the CRL on differently imbalanced training data. Metric: Mean *class-balanced accuracy* (%).

Network Architecture We evaluated the CRL in three state-of-the-art CNN models: (1) CifarNet [34], (2) ResNet32 [36], and (3) DenseNet [82]. Each CNN model was trained by the conventional CE loss (Eqn. (2)). The purpose is to test their performance gains in single-label object classification when incorporating the proposed CRL regularisation (Eqn. (8)).

Parameter Settings We trained each CNN from scratch with the learning rate at 0.1, the decay at 0.0005, the momentum at 0.9, the batch size at 256, and the epoch at 200. In the class *balanced* test, we cannot directly deploy our loss formulation (Eqn. (8)) as the imbalance measure $\Omega_{\text{imb}}=0$ hence eliminating the CRL. Instead, we integrated our CRL with the CE loss using equal weight by setting $\alpha=0.5$ for all models. For the class imbalanced cases, we set $\eta=0.01/0.5/0.5$ for CifarNet/ResNet32/DenseNet, respectively.

(I) Comparative Evaluation Table 8 shows the single-label object classification accuracy. Interestingly, it is found that our CRL approach can consistently improve state-of-the-art CNN models, e.g. increasing the accuracy of CifarNet/ResNet32/DenseNet by $3.6\%/1.2\%/0.8\%$, respectively. This shows that the advantages of our batch-wise minority class rectification method remain on class balanced cases. The plausible reasons are: (1) Whilst the global class distribution is balanced, random sampling of mini-batch adopted by common deep learning may introduce some imbalance in each iteration. Our per-batch balancing strategy hence has the chance to regularise inter-class margin and benefit the overall model learning. (2) The CRL considers the optimisation of class-level structural separation, which can provide a complementary benefit to the CE loss that instead performs per-sample single-class optimisation.

(II) Effect Analysis of Imbalanced Training Data We further evaluated the deep model performance and the CRL under different imbalance ratios. To this end, we carried out a controlled experiment by simulating class imbalance cases in training data. Specifically: **(1)** We simulated class imbalanced training data by a power-law class distribution as (Fig. 10(a)): $f_{\text{CS}}(i)=\frac{a}{i^{\gamma}+b}$, where $i\in\{1,2,\cdots,100\}$ is the class index, γ represents a preset parameter for controlling the imbalanced degree, a and b are two numbers estimated by the largest (500) and smallest (25) class size. We call the resulted training set "CIFAR-100$^{\text{imb}(\gamma)}$". **(2)** We constructed a corresponding dataset "CIFAR-100$^{\text{bln}(\gamma)}$", subject to having the same number of images covering all classes as "CIFAR-100$^{\text{imb}(\gamma)}$" and being class balanced (i.e. all classes are equally sized). This is necessary as "CIFAR-100$^{\text{imb}(\gamma)}$" and "CIFAR-100" differ in both data balance and size thus not directly comparable. **(3)** We trained the CNN models with and without CRL on these simulated training sets separately and tested their performances on the same standard test data. **(4)** To compare deep learning methods with conventional models, we also evaluated the k-nearest neighbour classifier with the HOG feature [83]. Table 9 shows the results when $\gamma=1$. We observed that: (1) Given class imbalanced training data, all three CNN models are adversely affected, with accuracy decreased by 3.9% (CifarNet), 6.1% (ResNet32), and 6.4% (DenseNet) respectively. Interestingly, the stronger CNNs suffer more performance degradation. (2) CRL improves all three CNN models by $2.0\%\sim3.2\%$ in accuracy, which show the effectiveness of CRL. (3) All three deep learning models are sensitive to imbalanced training data with similar relative performance drops as the conventional non-deep-learning HOG+kNN model. This suggests that deep learning models are not necessarily superior in tackling the class imbalanced learning challenge. Moreover, we evaluated the CRL with ResNet32 given different imbalance cases γ ranging from 0.2 to 1.0. Figure 10 (b) shows its performance gains across all these settings. We observed no clear trend between model performance and γ since their relationship is non-linear. In particular, the model generalisation depends not only on the class distribution but also on other factors such as the specific training samples, i.e. information content is variable (and unknown) over training samples.

TABLE 9
Effect of the CRL in different CNN models given class imbalanced training data ($\gamma=1$). Metric: Mean *class-balanced accuracy* (%).

Training Dataset	CifarNet	ResNet32	DenseNet	HOG+kNN
CIFAR-100$^{\text{bln}(1)}$	38.6	48.2	52.7	7.3
CIFAR-100$^{\text{imb}(1)}$	34.7	42.1	46.3	6.5
	+CRL			
CIFAR-100$^{\text{imb}(1)}$	**36.7**	**45.2**	**49.5**	N/A

4.4 Further Evaluations and Discussions

We conducted component analysis for providing more insights on CRL. By default, we adopted the class-level relative comparison based CRL (Eqn. (9)) and used the most imbalanced X-Domain dataset, unless declared otherwise.

TABLE 10
Comparing hard mining schemes (Class/Instance) and CRL loss functions (Relative (**Rel**), Absolute (**Abs**), and Distribution (**Dis**)). Metric: Gain in the mean *class-balanced accuracy* (%).

Dataset	CelebA [12]			X-Domain [11]		
Loss Design	Abs	Rel	Dis	Abs	Rel	Dis
Instance Level	5.30	5.45	3.45	4.90	6.46	2.87
Class Level	5.08	**6.32**	4.90	4.76	**6.89**	4.32

CRL Design We evaluated the two hard mining schemes (*class-level* and *instance-level*, Sec. 3.2), and three loss types (*relative*, *absolute*, and *distribution comparison*, Sec. 3.3). We tested therefore 6 CRL design combinations in the comparison with baseline models without imbalanced learning: "DeepID2" on CelebA and "MTCT" on X-domain. Table 10

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2832629, IEEE Transactions on Pattern Analysis and Machine Intelligence

shows that: (1) All CRL models improve the mean accuracy consistently, with the CRL(Class+Rel) the best. (2) With the same loss type, the class-level design is superior in most cases. This suggests that regularising the score space is more effective than the feature space. A plausible explanation is that the former is more compatible with the conventional CE loss which also operates with class scores.

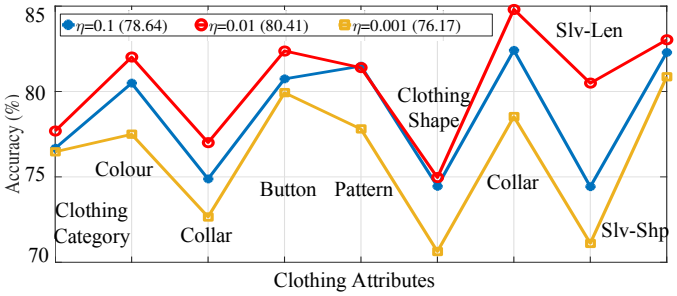


Fig. 11. Effect of the weight between the CE loss and our CRL on X-domain by varying η in Eqn. (8). Metric: *Class-balanced accuracy* (%). The mean accuracy for each setting is given in the parentheses.

Loss Weight Optimisation We evaluated the effectiveness of the weight between the CE loss and the CRL loss by tuning the coefficient η in Eqn. (8) on a range from 0.001 to 0.1 using the X-Domain benchmark. Figure 11 shows that the best weight selection is $\eta = 0.01$. Moreover, it is found that the change in η affects the performance on most or all attributes consistently. This indicates that the CRL formulation with loss weighting is imbalance adaptive, capable of effectively modelling multiple attribute labels with diverse class imbalance ratios by a single-value hyper-parameter (η) optimisation using cross-validation.

TABLE 11
Effect of the CRL hard mining (HM) and joint learning (JL) in comparison to the LMLE on X-Domain.

Method	CRL(HM+JL)	CRL(JL)	LMLE [17]
Mean Accuracy (%)	**80.42**	78.89	75.77

Hard Mining and Joint Learning We further evaluated the individual effects of Hard Mining (HM) and Joint Learning (JL) the features and classifier in the CRL model "CRL(HM+JL)", as in comparison to LMLE[8] [17]. Table 11 shows the performance benefit of the proposed CRL Hard Mining (HM) as compared to CRL without HM "CRL(JL)", with a 1.53% (80.42-78.89) mean accuracy advantage on X-Domain. It also shows that the joint learning in CRL has a mean accuracy advantage of 3.12% (78.89-75.77) over LMLE which has no joint learning but has hard mining.

Top-κ We examined the effect of different κ values in hard mining from 1 to 175 with step-size 25. Figure 12 shows that when κ=1 (i.e. hardest mining), the model fails to capture a good converging trajectory. This is because the hardest mining represents over sparse and possibly incorrect (due to outlier noise) class boundaries, which hence causes poorer optimisation. When $\kappa >= 25$, there is no further improvement to model learning. Given that larger κ increases the model training cost, we set $\kappa = 25$ for all our experiments.

8. In this evaluation, we treat LMLE as a whole without separating/removing its built-in hard mining mechanism.

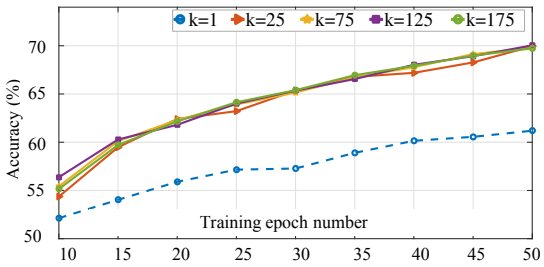


Fig. 12. Effect of κ (sample numbers) in hard mining on X-Domain.

TABLE 12
Effect of the CRL class scope on X-Domain.

CRL Class Scope	Mean Accuracy (%)	Training Time
Minority Classes	80.42	**60.4** Hours
All Classes	**81.30**	77.6 Hours

Class Scope We also evaluated the effect of the class scope (Minority-Classes and All-Classes) on which the CRL regularisation is enforced in terms of both accuracy and model training cost. Table 12 shows that applying the CRL to all classes in each batch yields superior performance. Specifically, relative to the baseline MTCT's 73.53% mean accuracy (Table 5), the scopes of Minority-Classes and All-Classes bring 6.89% (80.42-73.53) and 7.78% (81.30-73.53) accuracy gain, respectively. That is, the latter yields additional 12.9% ((7.78-6.89)/6.89) gain but at 28.5% ((77.6-60.4)/60.4) extra training cost. This suggests better cost-effectiveness by focusing *only* on minority classes in imbalanced data learning.

TABLE 13
Effect of minority class criterion (Eqn. (3)) on X-Domain.

Minority Class Criterion	ρ=10%	ρ=30%	ρ=50% (**Ours**)
Mean Accuracy (%)	77.82	79.29	**80.42**

Minority Class Criterion At last, we evaluated the effect of minority class criterion (ρ in Eqn. (3)) on a range from 10% to 50% to generalise the two-class minority class definition to a multi-class setting. Table 13 shows the effect on model performance when ρ changes, demonstrating that a minority class criterion setting with $\rho = 50\%$ is both most effective and conceptually consistent with the two-class setting.

5 Conclusion

In this work, we introduced an end-to-end class imbalanced deep learning framework for large scale visual data learning. The proposed Class Rectification Loss (CRL) approach is characterised by batch-wise incremental minority class rectification with a scalable hard mining principle. Specifically, the CRL is designed to regularise the inherently biased deep model learning behaviour given extremely imbalanced training data. Importantly, CRL preserves the model optimisation convergence characteristics of stochastic gradient descent, therefore allowing for efficient end-to-end deep learning on significantly imbalanced training data with multi-label semantic interpretations. Comprehensive experiments were carried out to show the clear advantages and scalability of the CRL method over not only the state-of-the-art imbalanced data learning models but also dedicated deep learning visual recognition methods. For example, the

CRL surpasses the best alternative LMLE by 3% on the CelebA facial attribute benchmark and 5% on the extremely imbalanced X-Domain clothing attribute benchmark, whilst enjoying over 7× faster model training advantage. Our experiments also show the benefits of the CRL in learning standard deep models given class balanced training data. Finally, we provided detailed component analysis for giving insights into the characteristics of the CRL model design.

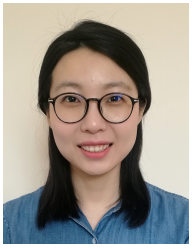
ACKNOWLEDGEMENTS

We shall thank Victor Lempitsky for providing the histogram loss code, Chen Huang and Chen Change Loy for sharing the pre-trained DeepID2 face recognition model. This work was partly supported by the China Scholarship Council, Vision Semantics Ltd., the Royal Society Newton Advanced Fellowship Programme (NA150459), and Innovate UK Industrial Challenge Project on Developing and Commercialising Intelligent Video Analytics Solutions for Public Safety.

REFERENCES

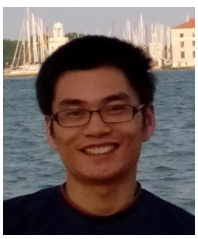
[1] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intelligent Data Analysis*, vol. 6, no. 5, pp. 429–449, 2002. 1, 2
[2] G. M. Weiss, "Mining with rarity: a unifying framework," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 7–19, 2004. 1, 2
[3] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE TKDE*, vol. 21, no. 9, pp. 1263–1284, 2009. 1, 2, 3, 5, 7, 8, 10
[4] T. M. Hospedales, S. Gong, and T. Xiang, "Finding rare classes: Active learning with generative and discriminative models," *IEEE TKDE*, vol. 25, no. 2, pp. 374–386, 2013. 1
[5] C. Drummond, R. C. Holte *et al.*, "C4.5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling," in *ICML Workshop*, 2003. 1, 2, 7, 8, 10
[6] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *JAIR*, vol. 16, pp. 321–357, 2002. 1, 2
[7] T. Maciejewski and J. Stefanowski, "Local neighbourhood extension of smote for mining imbalanced data," in *ICDM*, 2011. 1, 2
[8] K. M. Ting, "A comparative study of cost-sensitive boosting algorithms," in *ICML*, 2000. 1, 2
[9] Y. Tang, Y.-Q. Zhang, N. V. Chawla, and S. Krasser, "Svms modeling for highly imbalanced classification," *IEEE TSMCB*, vol. 39, no. 1, pp. 281–288, 2009. 1, 2
[10] R. Akbani, S. Kwek, and N. Japkowicz, "Applying support vector machines to imbalanced datasets," in *ECML*, 2004. 1, 2
[11] Q. Chen, J. Huang, R. Feris, L. M. Brown, J. Dong, and S. Yan, "Deep domain adaptation for describing people based on fine-grained clothing attributes," in *CVPR*, 2015. 1, 2, 3, 7, 8, 9, 10, 11
[12] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *ICCV*, 2015. 1, 2, 3, 7, 8, 11
[13] B. Krawczyk, "Learning from imbalanced data: open challenges and future directions," *Progress in Artificial Intelligence*, vol. 5, no. 4, pp. 221–232, 2016. 1
[14] S. Gong, M. Cristani, S. Yan, and C. C. Loy, *Person re-identification*. Springer, 2014, vol. 1. 1
[15] R. Feris, R. Bobbitt, L. Brown, and S. Pankanti, "Attribute-based people search: Lessons learnt from a practical surveillance system," in *ICMR*, 2014. 1
[16] K. Chen, S. Gong, T. Xiang, and C. Loy, "Cumulative attribute space for age and crowd density estimation," in *CVPR*, 2013. 1
[17] C. Huang, Y. Li, C. Change Loy, and X. Tang, "Learning deep representation for imbalanced classification," in *CVPR*, 2016. 1, 2, 3, 4, 7, 8, 9, 10, 12
[18] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev, "Panda: Pose aligned networks for deep attribute modeling," in *CVPR*, 2014, pp. 1637–1644. 1, 2, 3, 8
[19] I. Triguero, S. del Río, V. López, J. Bacardit, J. M. Benítez, and F. Herrera, "Rosefw-rf: the winner algorithm for the ecbdl14 big data competition: an extremely imbalanced big data bioinformatics problem," *Knowledge-Based Systems*, vol. 87, pp. 69–79, 2015. 1
[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015. 2, 10
[21] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *CVPR*, 2014, pp. 806–813. 2
[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012, pp. 1097–1105. 2
[23] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE TPAMI*, vol. 35, no. 8, pp. 1798–1828, 2013. 2
[24] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *IEEE TKDE*, vol. 18, no. 1, pp. 63–77, 2006. 2, 3
[25] P. Jeatrakul, K. W. Wong, and C. C. Fung, "Classification of imbalanced data by combining the complementary neural network and smote algorithm," in *International Conference on Neural Information Processing*, 2010. 2, 3
[26] R. Alejo, V. García, J. M. Sotoca, R. A. Mollineda, and J. S. Sánchez, "Improving the classification accuracy of rbf and mlp neural networks trained with imbalanced samples," in *International Conference on Intelligent Data Engineering and Automated Learning*, 2006. 2, 3
[27] T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "Supervised neural network modeling: an empirical investigation into learning from imbalanced data with labeling errors," *IEEE TNN*, vol. 21, no. 5, pp. 813–830, 2010. 2, 3
[28] M. A. Mazurowski, P. A. Habas, J. M. Zurada, J. Y. Lo, J. A. Baker, and G. D. Tourassi, "Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance," *Neural Networks*, vol. 21, no. 2, pp. 427–436, 2008. 2, 3
[29] J. Huang, R. S. Feris, Q. Chen, and S. Yan, "Cross-domain image retrieval with a dual attribute-aware ranking network," in *ICCV*, 2015. 2, 3, 9, 10
[30] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "Deepfashion: Powering robust clothes recognition and retrieval with rich annotations," in *CVPR*, 2016. 2, 3, 9, 10
[31] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014. 2
[32] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *IJCV*, vol. 115, no. 3, pp. 211–252, 2015. 2, 9
[33] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *IJCV*, vol. 111, no. 1, pp. 98–136, 2015. 2
[34] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," 2009. 2, 7, 8, 10, 11
[35] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," 2007. 2
[36] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016. 3, 10, 11
[37] Q. Dong, S. Gong, and X. Zhu, "Multi-task curriculum transfer deep learning of clothing attributes," in *WACV*, 2017. 2, 3, 7, 9, 10
[38] D. Arthur and S. Vassilvitskii, "How slow is the k-means method?" in *ACM Annual Symposium on Computational Geometry*, 2006, pp. 144–153. 2
[39] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *CVPR*, 2015. 2, 3, 6, 8
[40] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-smote: a new over-sampling method in imbalanced data sets learning," in *International Conference on Intelligent Computing*, 2005. 2
[41] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *CVPR*, 2014. 2, 3
[42] N. Japkowicz *et al.*, "Learning from imbalanced data sets: a comparison of various strategies," in *AAAI Workshop*, 2000. 2
[43] R. Barandela, J. S. Sánchez, V. Garcıa, and E. Rangel, "Strategies for learning in class imbalance problems," *Pattern Recognit.*, vol. 36, no. 3, pp. 849–851, 2003. 2

[44] C. Chen, A. Liaw, and L. Breiman, "Using random forest to learn imbalanced data," *University of California, Berkeley*, 2004. 2
[45] Y. Lin, Y. Lee, and G. Wahba, "Support vector machines for classification in nonstandard situations," *Machine Learning*, vol. 46, no. 1, pp. 191–202, 2002. 2
[46] B. Liu, Y. Ma, and C. K. Wong, "Improving an association rule based classifier," in *European Conference on Principles of Data Mining and Knowledge Discovery*, 2000. 2
[47] B. Zadrozny and C. Elkan, "Learning and making decisions when costs and probabilities are both unknown," in *SIGKDD*, 2001, pp. 204–213. 2
[48] J. R. Quinlan, "Improved estimates for the accuracy of small disjuncts," *Machine Learning*, vol. 6, no. 1, pp. 93–98, 1991. 2
[49] G. Wu and E. Y. Chang, "Kba: Kernel boundary alignment considering imbalanced data distribution," *IEEE TKDE*, vol. 17, no. 6, pp. 786–795, 2005. 2
[50] B. Zadrozny, J. Langford, and N. Abe, "Cost-sensitive learning by cost-proportionate example weighting," in *ICDM*, 2003. 2
[51] F. Provost, "Machine learning from imbalanced data sets 101," in *AAAI*, 2000. 2
[52] B. Krawczyk and M. Woźniak, "Cost-sensitive neural network with roc-based moving threshold for imbalanced classification," in *International Conference on Intelligent Data Engineering and Automated Learning*, 2015. 2
[53] H. Yu, C. Sun, X. Yang, W. Yang, J. Shen, and Y. Qi, "Odoc-elm: Optimal decision outputs compensation-based extreme learning machine for classifying imbalanced data," *Knowledge-Based Systems*, vol. 92, pp. 55–70, 2016. 2
[54] J. Chen, C.-A. Tsai, H. Moon, H. Ahn, J. Young, and C.-H. Chen, "Decision threshold adjustment in class prediction," *SAR and QSAR in Environmental Research*, vol. 17, no. 3, pp. 337–352, 2006. 2, 7, 8, 10
[55] M. Wozniak, *Hybrid classifiers: methods of data, knowledge, and classifier combination*. Springer, 2013, vol. 519. 3
[56] S. Wang, Z. Li, W. Chao, and Q. Cao, "Applying adaptive oversampling technique based on data density and cost-sensitive svm to imbalanced learning," in *IJCNN*, 2012. 3
[57] M. Woźniak, M. Graña, and E. Corchado, "A survey of multiple classifier systems as hybrid systems," *Information Fusion*, vol. 16, pp. 3–17, 2014. 3
[58] J. Lan, M. Y. Hu, E. Patuwo, and G. P. Zhang, "An investigation of neural network classifiers with unequal misclassification costs and group sizes," *Decision Support Systems*, vol. 48, no. 4, pp. 582–591, 2010. 3
[59] Y.-M. Huang, C.-M. Hung, and H. C. Jiau, "Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem," *Nonlinear Analysis: Real World Applications*, vol. 7, no. 4, pp. 720–747, 2006. 3
[60] F. Fernández-Navarro, C. Hervás-Martínez, and P. A. Gutiérrez, "A dynamic over-sampling procedure based on sensitivity for multi-class problems," *Pattern Recognit.*, vol. 44, no. 8, pp. 1821–1833, 2011. 3, 7
[61] C. L. Castro and A. P. Braga, "Novel cost-sensitive approach to improve the multilayer perceptron performance on imbalanced data," *IEEE TNNLS*, vol. 24, no. 6, pp. 888–899, 2013. 3
[62] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, and R. Togneri, "Cost-sensitive learning of deep feature representations from imbalanced data," *IEEE TNNLS*, 2017. 3
[63] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang, "Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection," in *CVPR*, 2015. 3
[64] S. Wang, W. Liu, J. Wu, L. Cao, Q. Meng, and P. J. Kennedy, "Training deep neural networks on imbalanced data sets," in *IJCNN*, 2016. 3
[65] S. Guan, M. Chen, H.-Y. Ha, S.-C. Chen, M.-L. Shyu, and C. Zhang, "Deep learning with mca-based instance selection and bootstrapping for imbalanced data classification," in *CIC*, 2015. 3
[66] Y. Yan, M. Chen, M.-L. Shyu, and S.-C. Chen, "Deep learning for imbalanced multimedia data classification," in *ISM*, 2015. 3
[67] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE TPAMI*, vol. 32, no. 9, pp. 1627–1645, 2010. 3
[68] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *CVPR*, 2016. 3
[69] H. Oh Song, Y. Xiang, S. Jegelka, and S. Savarese, "Deep metric learning via lifted structured feature embedding," in *CVPR*, 2016. 3
[70] X. Wang and A. Gupta, "Unsupervised learning of visual representations using videos," in *ICCV*, 2015. 3
[71] E. Ustinova and V. Lempitsky, "Learning deep embeddings with histogram loss," in *NIPS*, 2016. 3, 5, 6
[72] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, "Learning fine-grained image similarity with deep ranking," in *CVPR*, 2014. 3
[73] T.-Y. Liu, "Learning to rank for information retrieval," *Foundations and Trends in Information Retrieval*, vol. 3, no. 3, pp. 225–331, 2009. 3, 5, 6
[74] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *CVPR*, 2005. 3, 5, 6
[75] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. 4
[76] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv*, 2015. 7
[77] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *NIPS*, 2014. 8, 9
[78] T. Evgeniou and M. Pontil, "Regularized multi–task learning," in *SIGKDD*, 2004, pp. 109–117. 8
[79] R. K. Ando and T. Zhang, "A framework for learning predictive structures from multiple tasks and unlabeled data," *JMLR*, vol. 6, no. Nov, pp. 1817–1853, 2005. 8, 9
[80] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008. 9
[81] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv*, 2013. 9
[82] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, "Densely connected convolutional networks," in *CVPR*, 2017. 11
[83] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005. 11



Qi Dong received her B.Sc. (2012) from Tianjin University of Technology and M.Sc. (2015) from Sichuan University, respectively. She is a Ph.D student in the Queen Mary University of London. Her research interests include computer vision and deep learning.

Shaogang Gong is Professor of Visual Computation at Queen Mary University of London (since 2001), a Fellow of the Institution of Electrical Engineers and a Fellow of the British Computer Society. He received his D.Phil (1989) in computer vision from Keble College, Oxford University. His research interests include computer vision, machine learning and video analysis.



Xiatian Zhu is a Computer Vision Researcher at Vision Semantics Ltd. He received his Ph.D. from Queen Mary University of London. He won The Sullivan Doctoral Thesis Prize 2016, an annual award representing the best doctoral thesis submitted to a UK University in the field of computer vision. His research interest is computer vision.