# Range Loss for Deep Face Recognition with Long-Tailed Training Data

Xiao Zhang[1,2] Zhiyuan Fang[1,3] Yandong Wen[4] Zhifeng Li[5] Yu Qiao*[1,6]

[1] Guangdong Provincial Key Laboratory of Computer Vision and Vitrual Reality Technology,
Shenzhen Institutes of Advanced Technology, Chinese Academy of Science, China

[2] Tianjin University [3] Southern University of Science and Technology [4] Carnegie Mellon University

[5] Tencent AI Lab, China [6] The Chinese University of Hong Kong, Hong Kong SAR, China

zhangx9411@gmail.com fangzy@mail.sustc.edu.cn yandongw@andrew.cmu.edu
michaelzfli@tencent.com yu.qiao@siat.ac.cn

## Abstract

*Deep convolutional neural networks have achieved significant improvements on face recognition task due to their ability to learn highly discriminative features from tremendous amounts of face images. Many large scale face datasets exhibit long-tail distribution where a small number of entities (persons) have large number of face images while a large number of persons only have very few face samples (long tail). Most of the existing works alleviate this problem by simply cutting the tailed data and only keep identities with enough number of examples. Unlike these work, this paper investigated how long-tailed data impact the training of face CNNs and develop a novel loss function, called range loss, to effectively utilize the tailed data in training process. More specifically, range loss is designed to reduce overall intrapersonal variations while enlarge interpersonal differences simultaneously. Extensive experiments on two face recognition benchmarks, Labeled Faces in the Wild (LFW) [11] and YouTube Faces (YTF) [33], demonstrate the effectiveness of the proposed range loss in overcoming the long tail effect, and show the good generalization ability of the proposed methods.*

## 1. Introduction

Recent years witnessed the remarkable progresses of applying deep learning models in various computer vision tasks such as classification [14, 26, 29, 10, 9] , scene understanding [37, 36], and action recognition [13]. As for face recognition, deep CNNs like DeepID2+ [28], FaceNet [24], DeepFace [30], Deep FR [21], exhibit excellent performance, which even surpass human recognition ability at certain dataset such as LFW [11].

To train an effective deep face model, abundant training data [4] and well-designed training strategies are indispensable. Unlike large scale datasets like ImageNet [22] where each category contains almost the same number of images, most large scale face datasets exhibit long-tailed distribution, that is, only limited number of classes (persons) appear frequently, while most of the other classes have spares examples. This fact is shown in Figure 1, which illustrates the distribution of MS-Celeb-1M. Only a small number of persons have large number face images, while many persons have very few examples. Empirical studies and analysis show that classes with more samples will pose greater impact on the feature learning procedure [38, 22] and inversely cripple the models ability on tailed part. As a result, the model trained under an extremely imbalanced distributed dataset is lean to overfit the rich classes with large samples, and spare samples from poor classes tends to exhibit large intra-class dispension after training. Clearly, this fact will harm recognition performance of the final model.

Most previous works handled this problem by removing the samples from poor classes to achieve the class balance of training dataset. According to [20], the performance can improve slightly if one just preserves 40% of positive samples to make the training samples more uniform. However, such disposal strategys flaw is obvious: to simply abandon the data partially, information contained in these data may also be omitted. Poor classes can include complementary knowledge to rich classes which can boost the performance of the final models.

This paper addresses the long tail problem in the context of deep face recognition from two aspects. Note here “long tail” is used to describe the imbalanced distribution of training set, of which most identities have rare samples while a few identities have most samples. This is different from the classical definition of heavy-tailed distributions in probability theory[1]. Firstly, we deeply investigate how long tailed data impacts current deep CNN models for face recognition. Secondly, we propose a new loss function, namely

*The corresponding author: yu.qiao@siat.ac.cn

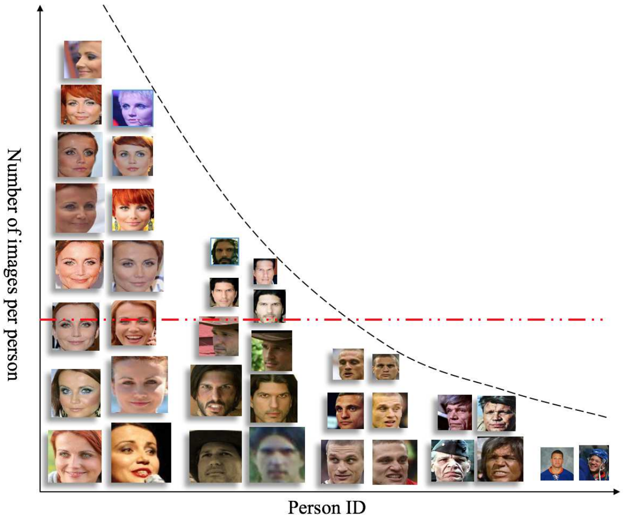

Figure 1. Long tail distributed data set of human faces, examples selected from MS-Celeb-1M [6]. Number of face images per person falls drastically, and only a small part of persons have large number of images. Cutting line in red represents the average number of images per person. Best viewed in color.

range loss to effectively enhance the model's learning ability towards tailed data. Specifically, this loss identifies the maximum Euclidean distance between all sample pairs as the range of this class. During training process, range loss encourages to minimize the range of each class for each mini-batch while preserves inter-class range large. To the best of our knowledge, this is the first work in the literature to discuss and address the long tail problem for deep face recognition models.

The main contributions of this paper are summarized as follows:

1) We empirically find that popular training losses of deep face recognition, i.e. contrastive loss [27], triplet loss [24], and center loss [32], all suffer from long tail distributions, while removing long tailed data can improve the recognition performance.

2) Extensive experiments and analysis on two famous face benchmarks (LFW [11] and YTF [33]) demonstrate the proposed range loss can largely relieve the long-tail effect and achieve superior performance than previous methods.

## 2. Related Work

Deep neural networks with great ability to learning representation from data, achieve remarkable successes in a series of vision tasks like recognition and detection [7, 25, 16, 8, 29], face recognition [21, 24, 27, 3, 34, 19, 31]. By increasing the depth, VGG [26] and GoogLeNet [4] achieved significant improvements on ImageNet [22] and VOC Pascale dataset [5]. More recently, Residual Network exploits shortcut connections to ease the training of substantially deeper networks [9]. Deep architectures like DeepID2+ [27], FaceNet [24], DeepFace [30], Deep FR [21], significantly boost the face recognition performance than previous shallow models. Loss function is important to train powerful deep models. DeepID2 utilized both verification and identification loss to enhance the training of CNNs [28]. FaceNet further shows that triplet loss contributes to improve the performance. More recently, [32] proposed center loss which takes account of class-clusters in CNN training. Different from these loss functions, range loss is defined on a new measure to minimize the within-person variations of deep representations.

Long tailed distribution of the data has been studied in scene parsing [34], and zero-shot learning [19]. In a workshop talk 2015, Bengio described the long tail distribution as the enemy of machine learning [23] . [20] investigates several factors that influence the performance of object detection with long tailed distribution of samples. Their analysis and empirical results indicate that classes with more samples will pose greater impact on the feature learning. And it is better to make the sample number more uniform across classes. Our work differs with these works in two aspects. Firstly, we study long tailed distribution for deep face recognition where the number of categories (persons) is huge. Secondly, instead of introducing data sampling strategies to balance training data, we propose novel loss functions which allows to exploit tailed data to improve the recognition performance.

## 3. The Proposed Approach

In this section, we firstly elaborate our empirical investigation and analysis on the effect of long tailed training data for deep face recognition, with VGG [26] and AlexNet [14] on LFW [11] and YTF [33] benchmarks. We also analyze the statistics of the deep representations learned with or without long tailed data. Based on these analysis, we propose a new loss function namely, range loss, to improve model robustness toward highly imbalanced dataset.

### 3.1. Problem formulation

In statistics, a long tail refers to the portion of a distribution having a large number of occurrences far from the "head" or central part of the distribution [2]. To investigate the long-tail property thoroughly in the context of deep learning face recognition, we first constructed a long tail distributed training set from MS-Celeb-1M [6] data set, which consists of 1.7 million face images with almost 100k identities. In this set, there are 700k images for roughly 10k identities, and 1 million images for the remaining 90k identities. The distribution of our training data is illustrated in Figure 2. We further divide the dataset into several groups according to the proportions of tailed data in Table 1. As shown in Figure 2, identities that contain less than 20 face

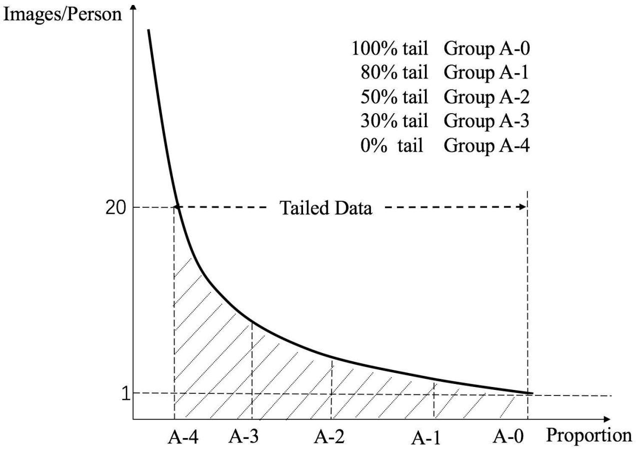

Figure 2. Our constructed data set with long-tailed distributions. The cutting lines represent the division proportions of tailed data.

| Groups | Num of Identities | Images | Tail Ratio |
|---|---|---|---|
| A-0 | 99,891 | 1,687,691 | 100.00% |
| A-1 | 81,913 | 1,620,526 | 80.00% |
| A-2 | 54,946 | 1,396,414 | 50.00% |
| A-3 | 26,967 | 1,010,735 | 30.00% |
| A-4 | 10,000 | 699,832 | 0.00% |

Table 1. Training Set with Long-tail Distribution. Control group's division proportion can be viewed in Fig. 2

images are regarded as poor classes (tailed data). In Table1, group A-0 contains all tailed data while A-4 includes no tailed data. A-1, A-2, A-3 have tailed data ratios as 80%, 50%, 30%, respectively.

We train the popular VGG-Face [14] with softmax cross-entropy loss, and then examine their performances on LFW [11] and YTF [33] s tasks. The accuracy are compared in Table 2. Training with A-1 and A-2 (with 80% and 50% tailed data) leads to better performance than training with A-0, even A-0 has more training examples than A-1 and A-2. On the other hand, if we remove too much tailed data like A-3 and A-4, the performance drops. These facts indicate the long tailed data can harm the training of deep face model, but it might not be good idea to remove all tailed data, since they may contribute too.

The above experiments are conducted using VGG-Face net with relatively large dataset. It might be interested to explore shallower networks with smaller datasets. We selected the AlexNet [14] as basic architecture and constructed new dataset as follows. We random sampling 4,000 identities from almost 100,000 identities of A-0 and use their images to construct dataset B-0. B-0 has similar distribution as A-0. By removing 1000, 2000 and 3000 identities from the tailed part of B-0, we get training sets B-1, B-2 and B-3. Then we train AlexNet [14] on these four training sets. The results of LFW [11] are shown in Table 3. B-3 has the highest accuracy with least number of identities while B-0 receives the lower result. These facts again verify the negative effect of the long-tailed data, especially for shallow networks with small training sets.

| Groups | Acc. on LFW | Acc. on YTF |
|---|---|---|
| A-0 (100% tail) | 97.87% | 92.0% |
| A-1 (80% tail) | 98.03% | 92.2% |
| A-2 (50% tail) | 98.25% | 92.9% |
| A-3 (30% tail) | 97.18% | 91.7% |
| A-4 (0% tail) | 95.97% | 89.4% |

Table 2. Performances comparison of softmax loss on LFW and YTF with/without long-tail data. VGG Net is used.

| Groups | Acc. on LFW |
|---|---|
| B-0 (4000, with long-tail) | 78.73% |
| B-1 (3000 identities) | 79.57% |
| B-2 (2000 identities) | 81.52% |
| B-3 (1000 identities) | 83.77% |

Table 3. Performances comparison of softmax loss on LFW with/without long-tail data. AlexNet is used. Since AlexNet has fewer layers and weights than VGG Net, its baseline is low, which makes long tail effect more obvious.

### 3.2. Explorations with contrastive, triplet and Center Losses

Recent studies has demonstrated that contrastive loss [27], triplet loss [24], and center loss [32] can help to improve the face recognition performance of deep CNNs. Contrastive loss [27] help to discriminate between positive face pairs of the same person and negative face pairs from different persons. Triplet loss [24] aims to minimize the distance between an anchor and a positive sample and maximize the distance between the anchor and a negative. Center loss [32] keeps the average feature vector (center) for every identity and optimizes the distance between the center and its associated feature vectors. Considering the characteristics of long tailed distributions, a small number of generic objects/entities appear very often while most others present more rarely. So there is a question whether these loss functions help to relieve the effect of tailed data.

We apply contrastive loss, triplet loss, and center loss together with softmax on VGG-16 with the same long tailed distributed datasets constructed in Section 3.1. We examine the face verification performance of the trained models on LFW and YTFs. As for the training pairs of contrastive and triplet loss, we divide the dataset into two parts with the same number of identities firstly. Positive pairs are randomly selected from the former part and negative pairs are generated in the latter part. The results are summarized in Table 4. Although these results are higher than those of soft-

| Groups | Contrastive Loss | | Triplet Loss | | Center Loss | |
|---|---|---|---|---|---|---|
| | Acc. on LFW | Acc. on YTF | Acc. on LFW | Acc. on YTF | Acc. on LFW | Acc. on YTF |
| A-0 (with long-tail) | **98.35%** | **92.7%** | **98.10%** | **92.3%** | **98.22%** | **92.4%** |
| A-1 (cut 20% tail) | 98.45% | 93.1% | 98.13% | 92.3% | 98.50% | 92.7% |
| A-2 (cut 50% tail) | **98.47%** | **93.3%** | **98.40%** | **93.2%** | **98.57%** | **93.2%** |
| A-3 (cut 70% tail) | 96.23% | 91.1% | 97.87% | 91.7% | 97.85% | 92.0% |
| A-4 (cut 100% tail) | 95.97% | 89.4% | 97.33% | 91.1% | 97.33% | 91.1% |

Table 4. Long-tail effect of contrastive Loss, triplet Loss and center loss. Eveluated on LFW and YTF with VGG Nets.

max loss only, long tail effect still exist. With 291,277 more tailed images than A-2, A-0 receives worse performance on LFW with accuracy down 0.12%, 0.30% and 0.3% for contrastive loss, triplet loss, and center loss, respectively. Similar tendency can be observed on YTF.

### 3.3. Analysis of Deep Feature Vectors

Well-trained CNN can map the input face image to feature vectors with rich identity information. For recognition tasks, we expect CNN model to output similar deep feature vectors for same persons and far apart vectors for different persons. In this subsection, we analyze the deep feature vectors calculated with and without tailed data. We randomly select 10 identities and 20 face images for each identity from the testing dataset. Each face image is mapped to a 4096-dimensional feature by VGG-Nets trained with above loss functions. Since it is hard to analyze high dimension vectors, we use t-SNE [18] to transform these vectors into 2-D vectors, as shown in Figure 3.

We also calculate the standard deviation (SD), average Euclidean metric (EM), and kurtosis of these 2-D feature vectors for intra-class and inter-class pairs. Kurtosis can be calculated by,

$$Kurt(\mathcal{X}) = \frac{m_4}{m_2^2} - 3 = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^4}{(\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2)^2} - 3, \qquad (1)$$

where $m_k$ is the $k^{th}$ central moment of $\mathcal{X}$ and the value of kurtosis is no less than $-2$.

Among these statistics, SD and EM represent sample variations, and Kurtosis indicates the existence of infrequent extreme samples. Thus it is expected that SD and EM are large for inter-class samples, while Kurtosis is small for intra-class samples. The statistics are summarized in Table 5. From these results, we firstly found long tailed dataset A-0 leads to larger Kurtosis than A-2 (fewer tailed data). This means that by removing tailed samples from training set, we can suppress extreme features in testing examples. Secondly, for inter-class evaluation, A-2 always exhibits larger SD and EM values than A-0 for all loss functions. This partly explains why training with A-2 leads to better testing performance than A-0. Similar conclusions can be drawn on results on B-0 to B-3. Finally, contrastive loss, triplet loss, and center loss although achieve higher accuracy, exhibit similar tendency as softmax loss. They cannot resolve the long-tail effect well.

Motivated by the above analysis, we find the necessity to propose novel loss function for handling training data with long tail distribution. Such loss function should be designed for better utilizing the infrequent extreme deviated data and preventing the increase of kurtosis, which we believe has been submerged by the richer classes information and poses negative impact in learning discriminative inter-class features.

### 3.4. The Range Loss

In long tail distributed data, samples of the tailed data are usually extremely rare, in other words, there are only very limited images for each person in the tailed parts. Contrastive loss optimizes the model in such a way that intra-class samples are pulled together and inter-class samples are pushed apart. In the training phase, it needs to construct positive pairs and negative pairs. It is difficult to obtain sufficient positive pairs of the same person on long tailed data. Moreover, as we discussed, richer classes will pose greater impact on the models training and may easily cause overfitting. Similar problems exist for triplet loss and center loss due to insufficient samples in the tailed parts.

This paper addresses this challenge by proposing range loss to handle imbalanced data. This new loss can help to reduce kurtosis (infrequent extreme deviation) while enlarge the inter-class distances. Inspired by contrastive loss, range loss penalizes intra-personal variations (especially for infrequent extreme deviated value) while enlarges the interpersonal differences simultaneously. Unlike contrastive loss defined on individual positive and negative pairs, range loss is defined on the overall distances between all sample pairs within one minibatch. In other words, we use statistics over the minibatch to define loss other than individual samples. Our range loss function is also partly inspired by the hard negative mining method which is widely used in the training of object detectors. Samples at classification boundaries deliver more important information for training classifiers. Here we use intra-class distance as a measure to identify these hard samples within one minibatch. In summary,

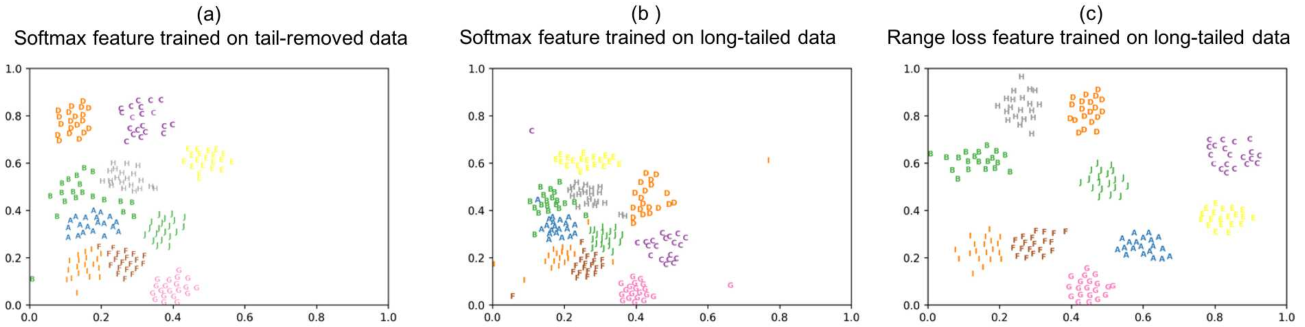

Figure 3. The fig (a) shows features trained with A-2 (50% tailed data) and softmax loss. The middle picture show features trained with full tailed data (A-0) and softmax loss. Both inter and intra class distances in middle picture are shorter than the left picture due to long tail impact. The fig (b) show features trained with full tailed data (A-0) and range loss, in which intra-samples locate nearly and intra-class distances are relatively large. Letters of the same color show samples of the same identity. Best viewed in color.

| Model | Loss | Data | **Intra-Class Evaluation** | | | **Inter-Class Evaluation** | | |
|---|---|---|---|---|---|---|---|---|
| | | | **SD** | EM | **Kurtosis** | **SD** | EM | Kurtosis |
| VggNet | Softmax | A-2 | 56.9768 | 77.9928 | -1.3590 | 285.6691 | 342.0850 | -1.4122 |
| | | A-0 | 22.8880 | 33.3013 | 0.3803 | 71.5123 | 88.4179 | -1.9884 |
| | Contras. | A-2 | 26.3160 | 36.4437 | -1.4130 | 122.3764 | 150.9392 | -1.7051 |
| | | A-0 | 22.8497 | 31.5918 | -1.0697 | 109.9323 | 134.6600 | -1.6276 |
| | Triplet | A-2 | 26.0807 | 36.7853 | -0.9714 | 113.1263 | 134.5524 | -1.0984 |
| | | A-0 | 23.0050 | 31.9569 | -0.9448 | 106.7124 | 129.6840 | -1.4492 |
| | Center | A-2 | 18.9627 | 25.7436 | -0.9558 | 180.4223 | 136.4760 | -1.2578 |
| | | A-0 | 15.2288 | 18.9850 | -0.5807 | 118.6627 | 92.2623 | -1.3320 |
| | **Range** | A-2 | 42.2881 | 63.4769 | **-1.3308** | 125.4162 | 153.2813 | -1.3468 |
| | | A-0 | 27.2868 | 39.5968 | **-1.4060** | 208.1743 | 249.8967 | -1.0050 |
| AlexNet | Softmax | B-3 | 341.7141 | 464.1585 | -1.5825 | 570.6800 | 685.8148 | -1.6795 |
| | | B-2 | 226.0212 | 308.4232 | -1.5487 | 474.1194 | 561.4882 | -1.7595 |
| | | B-1 | 217.6945 | 296.9807 | -1.2813 | 343.4455 | 415.3786 | -1.7131 |
| | | B-0 | 110.4576 | 149.8750 | -1.1898 | 177.3861 | 218.8029 | -1.9336 |
| | **Range** | B-3 | 240.3164 | 222.8551 | **-1.1285** | 541.4603 | 454.6582 | -1.2935 |
| | | B-2 | 227.1426 | 202.5412 | **-1.1413** | 476.1393 | 466.5233 | -1.5659 |
| | | B-1 | 177.6785 | 169.0733 | **-1.4545** | 525.6785 | 399.7669 | -1.3811 |
| | | B-0 | 142.5243 | 155.7600 | **-1.4412** | 486.4360 | 458.0701 | -1.5008 |

Table 5. The intra-class and inter-class statistics expose differences between long-tail model and cut-tail model. Here SD is standard deviation and EM is the average Euclidean metric. Good CNN models are expected to have small intra-class standard deviation and average Euclidean metric while large for inter-class. Kurtosis describes the $4^{th}$ order statistics of feature distribution. Infrequent extreme deviated vectors lead to high kurtosis. We always expect a low kurtosis because infrequent extreme deviation is harmful for face recognition task. Range loss resists the increase of kurtosis and restrains the extension of inter-class distance.

range loss should be designed to enlarge the distance among hard negative samples and thus lessen the largest intra-class variations. More specially, we identify the $k$ largest distances (ranges) of the intra-class pairs and use their harmonic mean value as a measure of intra-class loss. The final range value is determined by the intra-classs most distant sample pairs. For the inter-class loss, the shortest distance of class feature centers is used.

Mathematically, range loss can be formulated as,

$$\mathcal{L}_R = \alpha \mathcal{L}_{R_{intra}} + \beta \mathcal{L}_{R_{inter}}, \tag{2}$$

where $\alpha$ and $\beta$ are two weights, $\mathcal{L}_{R_{intra}}$ denotes the intra-class loss and $\mathcal{L}_{R_{inter}}$ represents the inter-class loss. $\mathcal{L}_{R_{intra}}$ penalizes the maximum harmonic range within each class:

$$\mathcal{L}_{R_{intra}} = \sum_{i \subseteq I} \mathcal{L}^i_{R_{intra}} = \sum_{i \subseteq I} \frac{k}{\sum_{j=1}^{k} \frac{1}{\mathcal{D}_j}}, \tag{3}$$

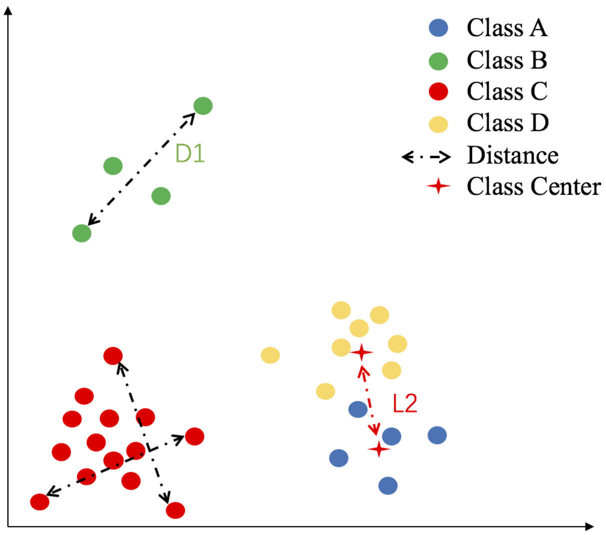

Figure 4. An illustration of rang loss for one minibatch in 2D space. There are 4 classes in this mini-batch, and Class B represents one typical tail-part class. $D_1$ denotes Class B's greatest intra-class distance. $L_2$ represents the center distance between Class D and Class A. The range loss is determined by the shortest center distances( $L_2$ in these 4 classes) and the harmonic mean value of the $k$ greatest ranges( $D_1$ as for Class B) in each class. (Best viewed in color).

where $I$ denotes the complete set of identities in current mini-batch, and $\mathcal{D}_j$ is the $j$-th largest distance. For example, let $\mathcal{D}_1 = \|x_1 - x_2\|_2^2$ and $\mathcal{D}_2 = \|x_3 - x_4\|_2^2$. $\mathcal{D}_1$ and $\mathcal{D}_2$ are the largest and second largest Euclidean range for a specific identity $i$ respectively. Input $x_1$ and $x_2$ denote two face samples with the longest distance, and similarly, input $x_3$ and $x_4$ are samples with the second longest distance. Equivalently, the overall cost is the harmonic mean of the first $k$-largest ranges within each class. Experiments show that $k = 2$ yields good performance. $k = 3$ has slightly improvement with $0.03\%$ for LFW on average but much hard to train while even get worse results for $k = 4$ and $k = 5$. Thus we set $k = 2$ without special notification.

$\mathcal{L}_{R_{inter}}$ represents the inter-class loss that

$$\begin{aligned}\mathcal{L}_{R_{inter}} &= \max(M - \mathcal{D}_{Center}, 0) \\ &= \max(M - \|\bar{x}_{\mathcal{Q}} - \bar{x}_{\mathcal{R}}\|_2^2, 0),\end{aligned} \tag{4}$$

where $\mathcal{D}_{Center}$ is the shortest distance between the centers of two classes, and $M$ is the max optimization margin of $\mathcal{D}_{Center}$. $\mathcal{Q}$ and $\mathcal{R}$ are the two nearest classes within the current mini-batch, while $\bar{x}_{Q}$ and $\bar{x}_{R}$ represents their centers.

In order to enlarge the discriminative ability at the same time, we use range loss joint with the softmax loss as the supervisory signals. The final loss function can be formulated as:

$$\mathcal{L} = \mathcal{L}_S + \lambda\mathcal{L}_R = -\sum_{i=1}^{M} \log \frac{e^{W_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_j^T x_i + b_j}} + \lambda\mathcal{L}_R, \tag{5}$$

where $\mathcal{L}_S$ is the softmax cross entropy loss function.

In the above expression, $M$ refers to the mini-batch size and $n$ is the number of identities within the training set. $x_i$ denotes the features of identity $y_i$ extracted by the last fully connected layers of CNN. $W_j$ and $b_j$ are the parameters of softmax function. Softmax loss and range loss are complementary to each other. Softmax is designed and widely used in classification task while range loss is introduced to reduce the long-tail effect. $\lambda$ is used as a scaler to balance the two supervisions. If $\lambda$ is set to 0, the overall loss function can be seen as the conventional softmax loss. In our experiments, the weights for softmax loss, intra class part, and inter class part of range loss are set as 1, $5 \times 10^{-5}$, and $1 \times 10^{-4}$, respectively. Because these parts of range loss have forms of Euclidean metrics, they should have lower weights compared with softmax loss, logarithms of probabilities.

According to the chain rule, gradients of the range loss with respect to $x_i$ can be computed as:

$$\frac{\partial \mathcal{L}_R}{\partial x_i} = \alpha \frac{\partial \mathcal{L}_{R_{intra}}}{\partial x_i} + \frac{\partial \mathcal{L}_{R_{inter}}}{\partial x_i} \tag{6}$$

For a specific identity, let $S = \sum_{i=1}^{k} \frac{1}{\mathcal{D}_i}$, $D_j$ is the distance $x_{j1}$ between $x_{j2}$. In Eq(8), $x_Q$ and $x_R$ are 2 classes that have the shortest distance between their center, we have

$$\frac{\partial \mathcal{L}_{R_{intra}}}{\partial x_i} = \frac{2k}{(D_j S)^2} \begin{cases} |x_{j1} - x_{j2}|, \text{ if } x_i = x_{j1} \\ |x_{j2} - x_{j1}|, \text{ if } x_i = x_{j2} \\ 0, \text{if } x_i \neq x_{j1}, x_{j2} \end{cases} \tag{7}$$

$$\frac{\partial \mathcal{L}_{R_{inter}}}{\partial x_i} = \begin{cases} \frac{\partial \mathcal{L}}{\partial x_Q} = \frac{1}{2n_R} \left| \frac{\sum x_R}{n_R} - \frac{\sum x_Q}{n_Q} \right|, \text{ if } x_i = x_Q \\ \frac{\partial \mathcal{L}}{\partial x_R} = \frac{1}{2n_Q} \left| \frac{\sum x_Q}{n_Q} - \frac{\sum x_R}{n_R} \right|, \text{ if } x_i = x_R \\ 0, \text{ if } x_i \neq x_Q, x_R \end{cases} \tag{8}$$

where $n_i$ denotes the number of samples in class $i$ of current mini-batch. And we summarize the computation of loss functions and gradients in Algorithm 1.

Range loss requires very few computational cost. The gradients of range loss can be calculated efficiently as shown in Eq(6), Eq(7) and Eq(8). We found experimentally that training with range loss only requires additional $0.2422\%$ time than that without range loss, which can be almost ignored in the training process.

### 3.5. Discussions on Range Loss's Effectiveness

Generally speaking, range loss adopts two stronger identifiability statistical parameters than contrastive loss and

**Algorithm 1** Calculate gradient for range loss

```
Require: Feature set $\{x_i\}$ extracted from the last fully connected layer. Hyper parameter $m$ and $\lambda$.
Ensure: The intra-class part of range loss $L_{R_{intra}}$ and the inter-class part of range loss $L_{R_{inter}}$. The gradient of intra-class $\frac{\partial L_{R_{intra}}}{\partial x_i}$ and inter-class $\frac{\partial L_{R_{inter}}}{\partial x_i}$.

  for each class $i \subseteq I$ in one mini-batch do
    Compute the arithmetic mean feature as feature center $c_i$ of class $i$.
    Compute the $k$ largest Euclidean distances $\{D_j\}$ among features $\{x_i\}$ of class $i$.
    Compute the harmonic mean of $\{D_j\}$ as the intra-class loss of class $i$, $L_R^i = \frac{k}{\sum_{j=1}^{k} \frac{1}{D_j}}$.
  end for
  Compute the intra-class loss $L_{R_{intra}} = \sum_{i \subseteq I} L_R^i = \sum_i \frac{k}{\sum_{j=1}^{k} D_j}$.
  Compute the intra-class gradient Eq(8).
  Compute the shortest distances $D_{center}$ among all feature centers $\{c_P\}$.
  if $M - D_{min} > 0$ then
    Output the inter-class gradient $\frac{\partial L_{R_{inter}}}{\partial x_i}$.
  else
    $\frac{\partial L_{R_{inter}}}{\partial x_i} = 0$.
  end if
```

others: distance of the peripheral points in the intra-class subspace, and the center distance of different classes. Both the range value and the center value is calculated based on samples with one minibatch. The mini-batch of range loss is constructed by selecting samples to balance the intra and inter class variations. Specially, each mini-batch has a size of 256, where we first randomly select 16 identities/classes and then select 16 images for each identity. Statistically speaking, range loss utilizes those training samples of one mini-batch in a joint way instead of individually (Softmax) or pairly (Contrastive), thus ensure the model's optimization direction comparatively balanced. Center loss calculates loss with one minibatch but lacks inter-class optimization. To give an intuitive explanations of the range loss, we have simulated a 2-D feature distribution graph in one mini-batch with 4 classes (see Fig. 4) and the real effect can be seen in the right of Fig. 3.

## 4. Experiments

In this section, we evaluate our range loss based models on two well known face verification benchmarks, LFW [11] and YTF [33] data sets. We firstly implemented our range loss with VGGs [26] architecture and train CNN models with 50% (A-2) and 100% (A-0) tailed datasets constructed in Section 3.1. We also conducted experiments with the deep architecture proposed by [32] for fair comparison with center loss which achieved the state-of-art performances on LFW and YTF.

### 4.1. Performances on LFW and YTF Data sets

LFW [11] is a database of face photographs designed for unconstrained face recognition, which consists of more than 13,000 facial images collected from the web. YouTube face (YTF) database [33] is designed for studying the problem of unconstrained face recognition in videos, which contains 3,425 videos of 1,595 different people.

We implement our customized range loss layer using the Caffe [12]. For comparison, we trained CNN models under the supervision of softmax loss only, contrastive loss, joint triplet loss, center loss, and range loss, respectively (the last four are jointly used with softmax loss). The results are summarized in Table 6. As can be seen, when trained with long tailed dataset A-0, range loss clearly outperforms baseline model with softmax loss, from 97.87% to 98.63% in LFW and 92.0% to 93.50% in YTF. Contrary to the experimental results of softmax loss, contrastive loss and triplet loss with full tailed data leads to lower accuracy, while our range loss can effectively exploit the tailed data part to enhance the training, with accuracy increase by 0.18% in LFW and 0.3% in YTF from A-2 to A-0. Moreover, range loss can prevent kurtosis rising and extend the inter-class distance with long tailed data from the statistics in Table 5.

For fair comparison, we compared the performance of these loss without hard mining. We extra trained VGGNets on long-tailed data with the hard mining strategy of FaceNet. The results are 96.55% in LFW and 90.1% in YTF, which are still lower than range loss without hard mining, 98.13% in LFW and 90.07% in YTF under the same training data.

Range loss performs well with stronger nets like ResNets for long tail training. We train ResNets (Fig 5) on long-tailed data with and without range loss, and test these two models on BLUFER dataset[15]. ResNet with range loss obtained accuracies of 92.10% in verification (FAR=0.1%) and 63.69% in Open-Set Identification (Rank=1, FAR=1%) while version without range loss gets 90.03% and 60.02%, respectively.

These results indicate that range loss can largely relieve the negative effect caused by tailed data. One can even boost the performance of trained model with the tailed parts with our new loss.

### 4.2. Comparison with state-of-the-art methods

To further examine the ability of range loss, we utilize a residual CNN [9] for the next experiments, whose architecture is shown in Figure 5. Different from our previous practice, the model is trained under 1.5M filtered

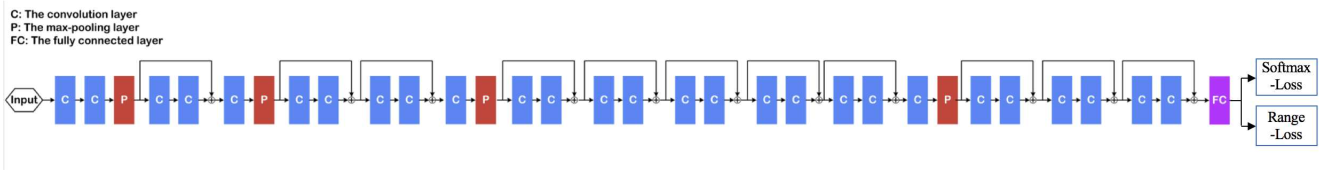

Figure 5. Residual Network's structure adopted in Section 4.2. The whole CNN is trained under the joint supervisory signals of soft-max and our range loss.

| Groups | Contrastive Loss | | Triplet Loss | | Center Loss | | **Range Loss** | |
|---|---|---|---|---|---|---|---|---|
| | LFW | YTF | LFW | YTF | LFW | YTF | LFW | YTF |
| A-0 (100% tail) | 98.35% | 92.7% | 98.10% | 92.3% | 98.22% | 92.4% | **98.63%** | **93.5%** |
| A-2 (50%) | 98.47% | 93.3% | 98.40% | 93.2% | 98.57% | 93.2% | **98.45%** | **93.2%** |

Table 6. Verification accuracy of Range Loss, Contrastive Loss, Triplet Loss, and Center Loss on LFW and YTF. A-0 contains all tailed data while A-2 includes 50% tailed data.

| Methods | LFW | YTF |
|---|---|---|
| DeepID-2+ [27] | 99.47% | 93.20% |
| FaceNet [24] | 99.63% | 95.10% |
| Baidu [17] | 99.13% | - |
| Deep FR [21] | 98.95% | 97.30% |
| DeepFace [30] | 97.35% | 91.40% |
| Center Loss [32] | 99.28% | 94.90% |
| Softmax Loss | 98.27% | 93.10% |
| Range Loss | **99.52%** | **93.70%** |

Table 7. Comparison with state of the art methods on LFW and YTF datasets.

data from MS-Celeb-1M [6] and CASIA-WebFace [35]. The intention of this experiment is to examine the potential ability and generalization of rang loss with deeper networks and cleaner data. We make comparison with a number of state-of-the-art methods, including DeepID-2+ [28], FaceNet [24], Baidu [17], DeepFace [30], and our residual net structure trained with softmax loss only. The results are given in Table 7. We have the following observations. Firstly, range loss again achieves better performance than baseline softmax with a clear margin (from 98.27% to 99.52% in LFW, and 93.10% to 93.70% in YTF). This indicates that the joint supervision of range loss and softmax loss can always enhance the deep networks ability to extract discriminative representations. Secondly, residual network integrated with range loss exhibits excellent performance on the two datasets and even outperforms most of previous famous networks. Although FaceNet has better performance than ours, it is trained on a super large datatsets, 133 times than ours.

## 5. Conclusions

In this paper, we deeply explore the effects the long tailed data in the context of training deep CNNs for face recognition. Contrary to our intuitiveness, long tailed data, if tailored properly, can contribute to boost the trained models performance. We propose a new loss function, namely range loss, to effectively exploit the tailed data in training deep networks. Our range loss contributes to reduce the intra-class variations and enlarge the inter-class distance for imbalanced and long tailed datasets. Experiments on two large scale face benchmarks, i.e. LFW and YTF, demonstrate the effectiveness of the proposed methods which clearly outperform baseline methods under long tailed conditions.

## 6. Acknowledge

This work was supported in part by National High-Tech Research and Development Program of China (2015AA042303), National Natural Science Foundation of China (U1613211), and External Cooperation Program of BIC Chinese Academy of Sciences (172644KYSB20160033).

## References

[1] S. R. Asmussen. Steady-state properties of gi/g/1. In *Stochastic Modelling and Applied Probability book series (SMAP, volume 51)*, pages 266–301. Springer, 2003. 1

[2] A. Bingham and D. Spradlin. The long tail of expertise. 2011. 2

[3] D. Chen, X. Cao, F. Wen, and J. Sun. Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3025–3032, 2013. 2

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009. 1, 2

[5] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010. 2

[6] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European Conference on Computer Vision*, pages 87–102. Springer, 2016. 2, 8

[7] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik. Learning rich features from rgb-d images for object detection and segmentation. In *European Conference on Computer Vision*, pages 345–360. Springer, 2014. 2

[8] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *European Conference on Computer Vision*, pages 346–361. Springer, 2014. 2

[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015. 1, 2, 7

[10] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1026–1034, 2015. 1

[11] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007. 1, 2, 3, 7

[12] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 7

[13] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014. 1

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1, 2, 3

[15] S. Liao, Z. Lei, D. Yi, and S. Z. Li. A benchmark study of large-scale unconstrained face recognition. In *Biometrics (IJCB), 2014 IEEE International Joint Conference on*, pages 1–8. IEEE, 2014. 7

[16] M. Lin, Q. Chen, and S. Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013. 2

[17] J. Liu, Y. Deng, and C. Huang. Targeting ultimate accuracy: Face recognition via deep embedding. *arXiv preprint arXiv:1506.07310*, 2015. 8

[18] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008. 4

[19] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*, 2013. 2

[20] W. Ouyang, X. Wang, C. Zhang, and X. Yang. Factors in finetuning deep model for object detection with long-tail distribution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 864–873, 2016. 1, 2

[21] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *British Machine Vision Conference*, volume 1, page 6, 2015. 1, 2, 8

[22] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 1, 2

[23] S.Bengio. The battle against the long tail. Talk on Workshop on Big Data and Statistical Machine Learning., 2015. 2

[24] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823, 2015. 1, 2, 3, 8

[25] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013. 2

[26] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1, 2, 7

[27] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *Advances in Neural Information Processing Systems*, pages 1988–1996, 2014. 2, 3, 8

[28] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2892–2900, 2015. 1, 2, 8

[29] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015. 1, 2

[30] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1708, 2014. 1, 2, 8

[31] Y. Wen, Z. Li, and Y. Qiao. Latent factor guided convolutional neural networks for age-invariant face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4893–4901, 2016. 2

[32] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, pages 499–515. Springer, 2016. 2, 3, 7, 8

[33] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 529–534. IEEE, 2011. 1, 2, 3, 7

[34] J. Yang, B. Price, S. Cohen, and M.-H. Yang. Context driven scene parsing with attention to rare classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3294–3301, 2014. 2

[35] D. Yi, Z. Lei, S. Liao, and S. Z. Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014. 8

[36] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object detectors emerge in deep scene cnns. *arXiv preprint arXiv:1412.6856*, 2014. 1

[37] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014. 1

[38] E. Zhou, Z. Cao, and Q. Yin. Naive-deep face recognition: Touching the limit of lfw benchmark or not? *arXiv preprint arXiv:1501.04690*, 2015. 1