Mis-Classified Vector Guided Softmax Loss for Face Recognition

Xiaobo Wang,[1*] Shifeng Zhang,[2*] Shuo Wang,[1] Tianyu Fu,[1] Hailin Shi,[1] Tao Mei[1]
[1]JD AI Research, Beijing, China
[2]CBSR & NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, China
wangxiaobo8@jd.com, shifeng.zhang@nlpr.ia.ac.cn, tmei@live.com

Abstract

Face recognition has witnessed significant progress due to the advances of deep convolutional neural networks (CNNs), the central task of which is how to improve the feature discrimination. To this end, several margin-based (*e.g.*, angular, additive and additive angular margins) softmax loss functions have been proposed to increase the feature margin between different classes. However, despite great achievements have been made, they mainly suffer from three issues: 1) Obviously, they ignore the importance of informative features mining for discriminative learning; 2) They encourage the feature margin only from the ground truth class, without realizing the discriminability from other non-ground truth classes; 3) The feature margin between different classes is set to be same and fixed, which may not adapt the situations very well. To cope with these issues, this paper develops a novel loss function, which adaptively emphasizes the mis-classified feature vectors to guide the discriminative feature learning. Thus we can address all the above issues and achieve more discriminative face features. To the best of our knowledge, this is the first attempt to inherit the advantages of feature margin and feature mining into a unified loss function. Experimental results on several benchmarks have demonstrated the effectiveness of our method over state-of-the-art alternatives. Our code is available at http://www.cbsr.ia.ac.cn/users/xiaobowang/.

Introduction

Face recognition is a fundamental and of great practice values task in the community of computer vision and pattern recognition. The task of face recognition contains two categories: face identification to classify a given face to a specific identity, and face verification to determine whether a pair of face images are of the same identity. Though it has been extensively studied for decades (Wang, Guo, and Li 2015; Hu et al. 2017; Liu, Hu, and Wang 2019; Hu et al. 2015; Wang et al. 2018e; 2019; Liu et al. 2019a; Sun, Wang, and Tang. 2015; Shi et al. 2017), there still exist a great many challenges for accurate face recognition, especially on large-scale test datasets at the very low false alarm rate (FAR), such as the MegaFace Challenge (Kemelmacher-Shlizerman et al. 2016; Nech and Kemelmacher 2017) and the recent Trillion-Pairs Challenge (Deepglint 2018).

In recent years, the advanced face recognition models are usually built upon deep convolutional neural networks (Wang et al. 2017b; He, Zhang, and Ren. 2016; Simonyan and Andrew 2014) and the learned discriminative features play a significant role. To train deep models, the CNNs are generally equipped with classification loss functions (Taigman, Yang, and Ranzato. 2014; Wen, Zhang, and Li 2016; Liang et al. 2017; Liu et al. 2017; Wang et al. 2018f), metric learning loss functions (Sun, Wang, and Tang. 2014; Schroff, Kalenichenko, and Philbin. 2015; Wang, Zhou, and Wen. 2017) or both (Sun, Wang, and Tang. 2015; Wen, Zhang, and Li 2016; Zheng, Pal, and Savvides 2018). Metric learning loss functions such as contrastive loss (Sun, Wang, and Tang. 2014) or triplet loss (Schroff, Kalenichenko, and Philbin. 2015) usually suffer from high computational cost. To avoid this problem, they require carefully designed sample mining strategies. But the performance is very sensitive to these strategies. So increasingly more researchers shift their attention to construct deep face recognition models by re-designing the classical classification loss functions.

Intuitively, face features are discriminative if their intra-class compactness and inter-class separability are well maximized. However, as pointed out by many recent studies (Wen, Zhang, and Li 2016; Wang et al. 2017a; Liu et al. 2017; Wang et al. 2018b; 2018f; Deng et al. 2019), the current prevailing classification loss function (*i.e.*, Softmax loss) lacks the power of feature discrimination for deep face recognition. To address this issue, Wen *et al.* (Wen, Zhang, and Li 2016) develop a center loss to learn centers for each identity to enhance the intra-class compactness. Wang *et al.* (Wang et al. 2017a) and Ranjan *et al.* (Ranjan, Castillo, and Chellappa. 2017) propose to use a scale parameter to control the temperature of softmax loss, producing higher gradients to the well-separated samples to shrink the intra-class variance. Recently, several margin-based softmax loss functions (Liu, Wen, and Yu 2016; Liu et al. 2017; Wang et al. 2018c; 2018b; Deng et al. 2019) to increase the feature margin between different classes have also been proposed. Liu *et al.* (Liu, Wen, and Yu 2016; Liu et al.

[*]These authors contributed equally to this work.

2017) introduce an angular margin (A-Softmax) between the ground truth class and other classes to encourage larger inter-class variance. However, it is usually unstable and the optimal parameters need carefully adjust for different settings. To enhance the stability of A-Softmax loss, Liang *et al.* (Liang et al. 2017) and Wang *et al.* (Wang et al. 2018b; 2018c) propose the additive margin (AM-Softmax) loss to stabilize the optimization. Deng *et al.* (Deng et al. 2019) develop an additive angular margin (Arc-Softmax) loss, which has a clear geometric interpretation.

Although the above approaches have achieved promising results, they mainly suffer from three shortcomings: 1) They obviously ignore the importance of informative features mining for discriminative learning. To address it, one may resort to the mining-based softmax loss functions. Shrivastava *et al.* (Shrivastava, Gupta, and Girshick. 2016) design the hard mining strategy (HM-Softmax) to improve the feature discrimination by constructing mini-batches using high-loss examples. But the percentage of hard examples is empirically decided and the easy examples are completely discarded. In contrast, Lin *et al.* (Lin, Goyal, and Girshick. 2017) design a relatively soft mining strategy, namely Focal loss (F-Softmax), to focus training on a sparse set of hard examples. However, the indication of hard examples is unclear. As a result, these two mining-based candidates usually fail to improve the performance. How to semantically select the hard examples is still an open problem. 2) They enlarge the feature margin only from the perspective of the ground truth class, which is partial and without realizing the discriminability from other non-ground truth classes. 3) Last but not at least, they enlarge the feature margin by using a same and fixed margin for all classes, which may not be appropriate and may not work very well in practice.

To overcome the aforementioned shortcomings, this paper tries to design a new loss function, which explicitly indicates the hard examples as mis-classified vectors and adaptively emphasizes on them to guide the discriminative feature learning. To sum up, the main contributions of this paper can be summarized as follows:

- We propose a novel MV-Softmax loss, which explicitly indicates the hard examples and focuses on them to guide the discriminative feature learning. As a consequence, our new loss also absorbs the discrimiantibility from other non-ground truth classes as well as is with adaptive margins for different classes.
- To the best of our knowledge, this is the first attempt to effectively inherit the merits of feature margin and feature mining techniques into a unified loss function. Moreover, We deeply analyze the relations and differences between our new loss and the current margin-based and mining-based losses.
- We conduct extensive experiments on the common benchmarks of LFW, CALFW, CPLFW, AgeDB, CFP, RFW, MegaFace and Trillion-Pairs, which have verified the superiority of our new approach over the baseline Softmax loss, the mining-based Softmax losses, the margin-based Softmax losses, and their naive fusions.

Preliminary Knowledge

Softmax. Softmax loss is defined as the pipeline combination of last fully connected layer, softmax function and cross-entropy loss. In face recognition, the weights $\boldsymbol{w}_k$, (where $k \in \{1, 2, \dots, K\}$ and K is the number of classes) and the feature $\boldsymbol{x}$ of the last fully connected layer are usually normalized and their magnitudes are replaced as a scale parameter s (Wang et al. 2017a; 2018b; Deng et al. 2019). In consequence, given an input feature vector $\boldsymbol{x}$ with its corresponding ground truth label y, the softmax loss can be reformulated as follows:

$$\mathcal{L}_1 = -\log \frac{e^{s\cos(\theta_{\boldsymbol{w}_y,\boldsymbol{x}})}}{e^{s\cos(\theta_{\boldsymbol{w}_y,\boldsymbol{x}})} + \sum_{k\neq y}^{K} e^{s\cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}})}}, \quad (1)$$

where $\cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}}) = \boldsymbol{w}_k^T\boldsymbol{x}$ is the cosine similarity and $\theta_{\boldsymbol{w}_k,\boldsymbol{x}}$ is the angle between $\boldsymbol{w}_k$ and $\boldsymbol{x}$. As pointed out by a great many studies (Liu, Wen, and Yu 2016; Liu et al. 2017; Wang et al. 2018b; Deng et al. 2019), the learned features with softmax loss are prone to be separable, rather than to be discriminative for face recognition.

Mining-based Softmax. Hard example mining is becoming a common practice to effectively train deep CNNs. Its idea is to concentrate on informative examples, thus it usually results in more discriminative features. There are recent works that select hard examples based on loss value (Shrivastava, Gupta, and Girshick. 2016; Lin, Goyal, and Girshick. 2017) to learn discriminative features. Generally, they can be summarized as:

$$\mathcal{L}_2 = -g(p_y)\log \frac{e^{s\cos(\theta_{\boldsymbol{w}_y,\boldsymbol{x}})}}{e^{s\cos(\theta_{\boldsymbol{w}_y,\boldsymbol{x}})} + \sum_{k\neq y}^{K} e^{s\cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}})}}, \quad (2)$$

where $p_y = \frac{e^{s\cos(\theta_{\boldsymbol{w}_y,\boldsymbol{x}})}}{e^{s\cos(\theta_{\boldsymbol{w}_y,\boldsymbol{x}})} + \sum_{k\neq y}^{K} e^{s\cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}})}}$ is the predicted ground truth probability and $g(p_y)$ is an indicator function. Basically, for the soft mining method Focal loss (Lin, Goyal, and Girshick. 2017) (F-Softmax), $g(p_y) = (1-p_y)^\gamma$, γ is a modulating factor. For the hard mining method HM-Softmax (Shrivastava, Gupta, and Girshick. 2016), $g(p_y) = 0$ when the sample is indicated as easy and $g(p_y) = 1$ when the sample is hard.

Margin-based Softmax. To directly enhance the feature discrimination, several margin-based softmax loss functions (Liu et al. 2017; Wang et al. 2018f; 2018b; Deng et al. 2019) have been proposed in recent years. In summary, they can be defined as follows:

$$\mathcal{L}_3 = -\log \frac{e^{sf(m,\theta_{\boldsymbol{w}_y,\boldsymbol{x}})}}{e^{sf(m,\theta_{\boldsymbol{w}_y,\boldsymbol{x}})} + \sum_{k\neq y}^{K} e^{s\cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}})}}, \quad (3)$$

where $f(m, \theta_{\boldsymbol{w}_y,\boldsymbol{x}})$ is a carefully designed margin function. Basically, $f(m_1, \theta_{\boldsymbol{w}_y,\boldsymbol{x}}) = \cos(m_1\theta_{\boldsymbol{w}_y,\boldsymbol{x}})$ is the motivation of A-Softmax loss (Liu et al. 2017), where $m_1 \geq 1$ and is an integer. $f(m_2, \theta_{\boldsymbol{w}_y,\boldsymbol{x}}) = \cos(\theta_{\boldsymbol{w}_y,\boldsymbol{x}}) - m_2$ with $m_2 > 0$ is the AM-Softmax loss (Wang et al. 2018b). $f(m_3, \theta_{\boldsymbol{w}_y,\boldsymbol{x}}) = \cos(\theta_{\boldsymbol{w}_y,\boldsymbol{x}} + m_3)$ with $m_3 > 0$ is the Arc-Softmax loss (Deng et al. 2019). More generally, the margin function can be summarized into a combined version: $f(m, \theta_{\boldsymbol{w}_y,\boldsymbol{x}}) = \cos(m_1\theta_{\boldsymbol{w}_y,\boldsymbol{x}} + m_3) - m_2$.

Problem Formulation

To begin with, let us retrospect the formulation of margin-based softmax losses, *i.e.*, Eq. (3), from which we can summarized that: 1) It ignores the importance of informative features mining for discriminative learning. 2) It only exploits the discriminability from the ground truth class y, *i.e*, $f(m, \theta_{\boldsymbol{w}_y,\boldsymbol{x}})$, without be aware of the potential discriminability from other non-ground truth classes k, where $k \neq y, k \in \{1, 2, \ldots, K\} \backslash \{y\}$. 3) It simply uses a same and fixed margin m_1, m_2 or m_3 to enlarge the feature margin between different classes.

Naive Mining-Margin Softmax Loss

To solve the first shortcoming, one may resort to hard examples mining strategies (Shrivastava, Gupta, and Girshick. 2016; Lin, Goyal, and Girshick. 2017). The mining-based loss functions aim to focus training on the hard examples while the margin-based loss functions are to enlarge the feature margin between different classes. Therefore, these two branches are orthogonal and can seamlessly incorporate into each other, leading a naive motivation to directly integrate them as:

$$\mathcal{L}_4 = -g(p_y) \log \frac{e^{sf(m,\theta_{\boldsymbol{w}_y,\boldsymbol{x}})}}{e^{sf(m,\theta_{\boldsymbol{w}_y,\boldsymbol{x}})} + \sum_{k\neq y}^{K} e^{s\cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}})}}. \quad (4)$$

The formulation Eq. (4) do involve informative features by the indicator function $g(p_y)$, but its improvement is limited in practice. The reason behind this may be, for the HM-Softmax (Shrivastava, Gupta, and Girshick. 2016), it explicitly indicates the hard examples, but it discards the easy ones. For the F-Softmax (Lin, Goyal, and Girshick. 2017), it uses all examples and empirically re-weights them by a modulating factor, but hard examples are unclear for training and without intuitive interpretation. This motivates us to design a more effective way to improve the performance.

Mis-classified Vector Guided Softmax Loss

Intuition says that considering the well-separated feature vectors has little effect on the learning problem. That means the mis-classified feature vectors are more crucial to enhance feature discriminability. To this end, we alternatively introduce a more elegant way to focus training on the truly informative features (*i.e.*, mis-classified vectors). Specifically, based on the margin-based softmax loss functions, we define a binary indicator I_k to adaptively indicate whether a sample (feature) is mis-classified by a specific classifier $\boldsymbol{w}_k$ (where $k \neq y$) in the current stage:

$$I_k = \begin{cases} 0, & f(m, \theta_{\boldsymbol{w}_y,x}) - \cos(\theta_{\boldsymbol{w}_k,x}) \geq 0 \\ 1, & f(m, \theta_{\boldsymbol{w}_y,x}) - \cos(\theta_{\boldsymbol{w}_k,x}) < 0 \end{cases}. \quad (5)$$

From the definition Eq. (5), we can see that if a sample (feature) is mis-classified, *i.e.*, $f(m, \theta_{\boldsymbol{w}_y,\boldsymbol{x}}) - \cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}}) < 0$ (*e.g.*, in the left sub-figure of Figure 1, the feature $\boldsymbol{x}_2$ belongs to class 1, but it is mis-classified by the classifier $\boldsymbol{w}_2$, *i.e.*, $f(m, \theta_{\boldsymbol{w}_1,\boldsymbol{x}_2}) - \cos(\theta_{\boldsymbol{w}_2,\boldsymbol{x}_2}) < 0$), it will be emphasized temporarily. In this way, the hard examples are explicitly indicated and we mainly focus on them for discriminative training. Consequently, we formulate our Mis-classified Vector guided Softmax (**MV-Softmax**) loss as follows:

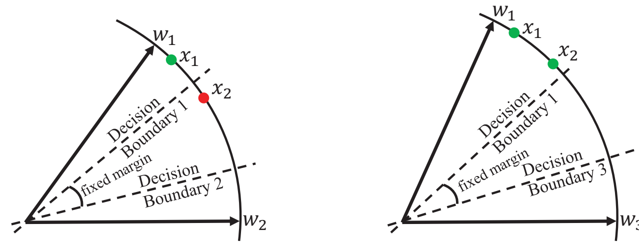

Figure 1: A geometrical interpretation of MV-Softmax from feature perspective. Samples $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are both from class 1. The mis-classified vectors (red dots) are those who are mis-classified by a specific classifier (*e.g.*, $\boldsymbol{w}_2$).

$$\mathcal{L}_5 = -\log \frac{e^{sf(m,\theta_{\boldsymbol{w}_y,\boldsymbol{x}})}}{e^{sf(m,\theta_{\boldsymbol{w}_y,\boldsymbol{x}})} + \sum_{k\neq y}^{K} h(t, \theta_{\boldsymbol{w}_k,\boldsymbol{x}}, I_k) e^{s\cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}})}}, \quad (6)$$

where $h(t, \theta_{\boldsymbol{w}_k,\boldsymbol{x}}, I_k) \geq 1$ is a re-weighted function to emphasize the indicated mis-classified vectors. Here we give two candidates, one is with fixed weights for all mis-classified classes:

$$h(t, \theta_{\boldsymbol{w}_k,\boldsymbol{x}}, I_k) = e^{stI_k}, \quad (7)$$

and the other one is an adaptive formulation:

$$h(t, \theta_{\boldsymbol{w}_k,\boldsymbol{x}}, I_k) = e^{st(\cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}})+1)I_k}. \quad (8)$$

where $t \geq 0$ is a preset hyperparameter. Obviously, when $t = 0$, the designed MV-Softmax loss Eq. (6) becomes identical to the original margin-based softmax losses Eq. (3).

Comparision to Mining-based Softmax Losses. To illustrate the advantages of our MV-Softmax loss over the traditional mining-based loss functions (*e.g.*, HM-Softmax (Shrivastava, Gupta, and Girshick. 2016) and F-Softmax (Lin, Goyal, and Girshick. 2017)), Figure 1 gives a toy example. Assume that we have two samples (features) $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, both of them are from class 1, where $\boldsymbol{x}_1$ is well-classified while $\boldsymbol{x}_2$ is not. The HM-Softmax empirically indicates the hard samples and discards the easy sample $\boldsymbol{x}_1$ to use the hard one $\boldsymbol{x}_2$ for training. The F-Softmax does not explicitly indicate the hard samples, but it re-weights all the samples, making the harder one $\boldsymbol{x}_2$ to have relatively larger loss value. These two strategies are directly from the loss viewpoint and the selection of hard examples is without semantic guidance. Our MV-Softmax loss Eq. (6) is from a different way. Firstly, we semantically indicates the hard examples (mis-classified vectors) according to the decision boundary. The hardness of previous methods is defined as a global relationship between feature (sample) and feature (sample). While our hardness is a local relationship between feature and classifier, which is more consistent with discriminative feature learning. Then, we emphasize these hard examples from probability viewpoint. Specifically, because the cross-entropy loss $-log(p)$ is a monotonically decreasing function, reducing the probability p (the reason is that $h(t, \theta_{\boldsymbol{w}_k,\boldsymbol{x}}, I_k) \geq 1$, see Eqs. (7) and (8)) of the mis-classified vector $\boldsymbol{x}_2$, will increase its importance for training. In summary, we can claim that our mis-classified vector guided mining strategy, is more superior for discriminative feature learning than previous ones.

Comparision to Margin-based Softmax Losses. Similarly, assume that we have a sample $\boldsymbol{x}_2$ from class 1, and it is not well-classified, (*e.g.*, the red dot in Figure 1). The original softmax loss aims to make $\boldsymbol{w}_1^T\boldsymbol{x}_2 > \boldsymbol{w}_2^T\boldsymbol{x}_2 \iff \cos(\theta_1) > \cos(\theta_2)$ and $\boldsymbol{w}_1^T\boldsymbol{x}_2 > \boldsymbol{w}_3^T\boldsymbol{x}_2 \iff \cos(\theta_1) > \cos(\theta_3)$. To make these objectives more rigorous, margin-based loss functions introduce a margin function $f(m, \theta_1) = \cos(m_1\theta_1 + m_3) - m_2$ from the perspective of ground truth class (*i.e.*, θ_1) (Liu et al. 2017; Wang et al. 2018b; Deng et al. 2019):

$$\begin{aligned} &\cos(\theta_1) \geq f(m, \theta_1) > \cos(\theta_2) \\ &\cos(\theta_1) \geq f(m, \theta_1) > \cos(\theta_3), \end{aligned} \tag{9}$$

wherein $f(m, \theta_1)$ is with a same and fixed margin for different classes and ignores the potential discriminability from other non-ground truth classes (*e.g.*, θ_2 and θ_3). To solve these issues, our MV-Softmax loss tries to further enlarge the feature margin from the perspective of other non-ground truth classes. Specifically, we have introduced a margin function $h^*(t, \theta_2)$ for the mis-classified feature $\boldsymbol{x}_2$:

$$\begin{aligned} &\cos(\theta_1) \geq f(m, \theta_1) > h^*(t, \theta_2) \geq \cos(\theta_2) \\ &\cos(\theta_1) \geq f(m, \theta_1) \geq \cos(\theta_3), \end{aligned} \tag{10}$$

where $h^*(t, \theta_2) = \log[h(t, \theta_2)e^{\cos(\theta_2)}] = \cos(\theta_2) + t$ or $(t + 1)\cos(\theta_2) + t$. For the case θ_3, because $\boldsymbol{x}_2$ is well-classified by the classifier $\boldsymbol{w}_3$, we do not need to give any additional enforcement to further enlarge its margin. Moreover, our MV-Softmax losses have also set adaptive margins for different classes. Taking MV-AM-Softmax (*i.e.*, $f(m, \theta_y) = \cos(\theta_y) - m$) as an example, for the mis-classified classes, the margin is $m + t$ or $m + t\cos(\theta_2) + t$. While for the well-classified classes, the margin is m. On account of these, our MV-Softmax losses have addressed the second and third shortcomings.

According to the above discussions, we conclude that our new loss has inherited the merits of feature margin and feature mining into a unified loss function, thus it is expected to achieve more discriminative features for face recognition.

Optimization

In this section, we show that our MV-Softmax loss Eq. (6) is trainable and can be easily optimized by the typical stochastic gradient descent (SGD). The difference between the previous margin-based softmax losses and the proposed MV-Softmax loss lies in the last fully connected layer $\boldsymbol{v} = [v_1, v_2, \ldots, v_K]^T = [\cos(\theta_{\boldsymbol{w}_1,\boldsymbol{x}}), \cos(\theta_{\boldsymbol{w}_2,\boldsymbol{x}}), \ldots, \cos(\theta_{\boldsymbol{w}_K,\boldsymbol{x}})]^T$. For the forward propagation, when $k = y$, it is the same as the original margin-based softmax loss (*i.e.*, $v_y = \cos(m_1\theta_{\boldsymbol{w}_y,\boldsymbol{x}} + m_3) - m_2$). When $k \neq y$, it has two cases, if the feature vector is well-classified for a specific classifier, it is the same as the original softmax (*i.e.*, $v_k = \cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}})$). Otherwise, it will be re-computed with a fixed weight $\cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}}) + t$ or an adaptive weight $(t + 1)\cos(\theta_{\boldsymbol{w}_k,\boldsymbol{x}}) + t$. The whole scheme of our method is summarized in Algorithm 1.

Algorithm 1: MV-Softmax

Input: Training set $\mathcal{S}$; The hyper-parameter t; Training epochs τ.
Initialization: $\alpha = 1$; Randomly initialize the parameter $\boldsymbol{\Theta}$ in convolution layers and $\boldsymbol{W}$ in the last fully connected layer.
while $\alpha \leq \tau$ **do**
  Shuffle the training set $\mathcal{S}$ and fetch mini-batch $\mathcal{S}_n$;
  Forward: According to the indication of hard examples Eq. (5), we compute the MV-Softmax loss by Eq. (6);
  Backward: Update the parameters $\boldsymbol{W}$ and $\boldsymbol{\Theta}$ by Stochastic Gradient Descent (SGD);
end
Output: Parameters $\boldsymbol{\Theta}$ and $\boldsymbol{W}$.

Table 1: Face datasets for training and test. "(P)" and "(G)" refer to the probe and gallery set, respectively.

	Datasets	Identities	Images
Training	MS-Celeb-1M-v1c-R	72,690	3.28M
Test	LFW	5,749	13,233
	CALFW	5,749	12,174
	CPLFW	5,749	11,652
	AgeDB	568	16,488
	CFP	500	7,000
	RFW	11,430	40,607
	MegaFace	530(P)	1M(G)
	Trillion-Pairs	5,749(P)	1.58M(G)

Experiments

Datasets

Training Data. The original MS-Celeb-1M dataset (Guo et al. 2016) contains about 100K identities with 10M images. However, it consists of a great many noisy faces. Fortunately, the trillion-pairs consortium (Deepglint 2018) has made their efforts to get a high-quality version MS-Celeb-1M-v1c, which is well-cleaned for training.

Test Data. We use eight face recognition benchmarks, including LFW (Huang, Ramesh, and Miller. 2007), CALFW (Zheng et al. 2017), CPLFW (Zheng et al. 2018), AgeDB (Moschoglou et al. 2017), CFP (Sengupta et al. 2016), RFW (Wang et al. 2018d), MegaFace (Kemelmacher-Shlizerman et al. 2016; Nech and Kemelmacher 2017) and Trillion-Pairs (Deepglint 2018), as the test data. For more details about the test datasets, please see their references.

Dataset Overlap Removal. In face recognition, it is very important to perform open-set evaluation, *i.e.*, there should be no overlapping identities between training set and test set. To this end, we need to carefully remove the overlapped identities between the employed training dataset (*i.e.*, MS-Celeb-1M-v1c) and the test datasets (including LFW, CALFW, CPLFW, AgeDB, CFP, RFW and MegaFace)[1]. For the overlap identities removal tool, we use the publicly available script provided by (Wang et al. 2018b) to check whether if two names are of the same person. As a consequence, we remove 14,186 identities from the training set MS-Celeb-

[1]For the Trillion-Pairs test set, we can not remove the potential overlaps because its ground truth label (name) is unreleased.

Table 2: Verification performance (%) of our MV-Softmax loss functions with different hyper-parameter t. 'f' and 'a' donate the fixed re-weight function Eq. (7) and the adaptive one Eq. (8), respectively.

Method	BLUFR 1e-5	CALFW	AgeDB
MV-Arc-Softmax-f (0.15)	94.60	**95.54**	98.05
MV-Arc-Softmax-f (0.2)	**95.18**	95.46	**98.11**
MV-Arc-Softmax-f (0.25)	94.04	95.51	98.08
MV-Arc-Softmax-a (0.25)	94.15	95.33	97.86
MV-Arc-Softmax-a (0.3)	**95.50**	95.46	**98.06**
MV-Arc-Softmax-a (0.35)	95.08	**95.50**	97.90
MV-AM-Softmax-f (0.2)	94.81	95.29	98.01
MV-AM-Softmax-f (0.25)	**95.74**	**95.45**	**98.05**
MV-AM-Softmax-f (0.3)	95.07	95.41	98.00
MV-AM-Softmax-a (0.15)	94.09	95.41	**98.13**
MV-AM-Softmax-a (0.2)	**96.27**	**95.63**	98.00
MV-AM-Softmax-a (0.25)	94.29	95.51	97.96

1M-v1c. For clarity, we donate the refined training dataset as MS-Celeb-1M-v1c-R. Important statistics of all the involved datasets are summarized in Table 1. To be rigorous, all the experiments in this paper are based on the refined training set MS-Celeb-1M-v1c-R. To encourage more researchers to abide by the open-set protocol, the overlapping lists and the refined dataset MS-Celeb-1M-v1c-R are publicly available.

Experimental Settings

Data Processing. We detect the faces by adopting the FaceBoxes detector (Zhang et al. 2017; 2019) and localize five landmarks (two eyes, nose tip and two mouth corners) through a simple 6-layer CNN (Feng et al. 2017; Liu et al. 2019b). The detected faces are cropped and resized to 144×144, and each pixel (ranged between [0,255]) in RGB images is normalized by subtracting 127.5 and divided by 128. For all the training faces, they are horizontally flipped with probability 0.5 for data augmentation.

CNN Architecture. In face recognition, there are many kinds of network architectures (Liu et al. 2017; Wang et al. 2018b; 2018a). To be fair, the CNN architecture should be the same to test different loss functions. As suggested by the work (Wang et al. 2018a), we use the Attention-Net (Wang et al. 2017b) to achieve a good balance between computation and accuracy. Moreover, inspired by the work (Deng et al. 2019), we integrate the IRSE module into the AttentionNet and rename the developed architecture as AttentionNet-IRSE. For the depth stages of AttentionNet-IRSE, we set [1, 1, 1] as our baseline architecture. The output of AttentionNet-IRSE gets a 512-dimension feature.

Training. All the CNN models are trained with stochastic gradient descent (SGD) algorithm and are trained from scratch, with the batch size of 32 on 4 P40 or 4 V100 GPUs parallelly, total batch size 128. The weight decay is set to 0.0005 and the momentum is 0.9. The learning rate is initially 0.1 and divided by 10 at 4, 8, 10 epochs, and we finish the training process at 12 epoch. All experiments in this paper are implemented by Pytorch library.

Test. At test stage, only the original image features are employed to compose the face representation. All the reported results in this paper are evaluated by a single model, without model ensemble or other fusion strategies.

For the evaluation metric, the cosine similarity is utilized. We follow the unrestricted with labelled outside data protocol (Huang, Ramesh, and Miller. 2007) to report the performance on LFW, CALFW, CPLFW, AgeDB, CFP and RFW. Moreover, we also report the BLUFR protocol (Liao et al. 2014) on the test set LFW. On Megaface and Trillion-Pairs Challenge, face identification and verification are conducted by ranking and thresholding the scores. Specifically, for face identification, the Cumulative Match Characteristics (CMC) curves are adopted to evaluate the Rank-1 accuracy. For face verification, the Receiver Operating Characteristic (ROC) curves are adopted. The true positive rate (TPR) at low false acceptance rate (FAR) is emphasized since in real applications false acceptance gives higher risks than false rejection.

For the compared methods, we compare our method with the baseline Softmax loss (**Softmax**) and the recently proposed state-of-the-arts, including 2 mining-based softmax losses (*i.e.*, **F-Softmax** and **HM-Softmax**), 3 margin-based softmax losses (**A-Softmax**, **Arc-Softmax** and **AM-Softmax**) and their 4 naive fusions (**F-Arc-Softmax**, **F-AM-Softmax**, **HM-Arc-Softmax** and **HM-AM-Softmax**). For all the competitors, their source codes can be downloaded from the github or from authors' webpages. The corresponding parameters of each competitors are mainly determined according to their paper's suggestions. Specifically, for HM-Softmax (Shrivastava, Gupta, and Girshick. 2016), we save 90% high-loss samples in each mini-batch for training. For F-Softmax, it is with the parameter $\gamma = 2.0$. For A-Softmax, the margin parameter is set as $m_1 = 3$. While for AM-Softmax and Arc-Softmax, the margin parameters are set as $m_2 = 0.35$ and $m_3 = 0.5$, respectively. The scale parameter s has already been discussed sufficiently in previous works (Wang et al. 2018b; 2018c). In this paper, we empirically fixed it to 32 for all the methods.

Exploratory Experiments

Effect of parameter t. Since the hyper-parameter t in the re-weighted function Eqs. (7) and (8) plays an important role in the developed MV-Softmax loss, we mainly explore to search its possible best value in this part. In Table 2, we list the performance of our proposed MV-Softmax loss function with t varies from different ranges. 'f' and 'a' donate the fixed re-weight function Eq. (7) and the adaptive one Eq. (8), respectively. From the numbers, we can summarize that our MV-Softmax loss is insensitive to the hyper-parameter t in a certain range. Moreover, according to this study, we empirically set $t = 0.2$ for MV-Arc-Softmax-f, $t = 0.3$ for MV-Arc-Softmax-a, $t = 0.25$ for MV-AM-Softmax-f and $t = 0.2$ for MV-AM-Softmax-a in the subsequent experiments.

Convergence of MV-Softmax. Although the convergence of our method is not easy to be theoretically analyzed, it would be intuitive to see its empirical behavior. Here, we give the loss changes as the number of epochs increases. From the curves in Figure 2, it can be observed that our

Table 3: Verification performance (%) of different loss functions on the test sets LFW, CALFW, CPLFW, AgeDB and CFP.

	Method	LFW	BLUFR			CALFW	CPLFW	AgeDB	CFP
			1e-3	1e-4	1e-5				
Baseline	Softmax	99.59	99.29	99.11	91.74	94.66	87.76	97.01	94.04
Mining-based	F-Softmax	99.65	99.24	98.72	91.19	93.83	86.35	96.51	93.20
	HM-Softmax	99.65	99.30	99.11	92.03	94.69	87.56	97.05	94.12
Margin-based	A-Softmax	99.65	99.30	99.12	92.77	94.55	87.85	97.16	94.22
	Arc-Softmax	99.76	99.33	99.30	93.75	95.44	88.78	98.00	95.28
	AM-Softmax	99.71	99.33	99.31	93.68	95.58	89.60	98.03	95.68
Naive-fused	F-Arc-Softmax	99.71	99.33	99.29	94.51	95.48	88.85	98.10	95.62
	F-AM-Softmax	99.73	99.33	99.30	92.81	95.58	89.60	**98.20**	95.47
	HM-Arc-Softmax	99.75	99.33	99.29	93.53	95.36	89.16	97.86	95.22
	HM-AM-Softmax	99.76	99.33	99.30	96.09	95.45	89.56	98.05	95.37
Ours	MV-Arc-Softmax-f (0.2)	99.78	**99.34**	99.30	95.18	95.46	89.30	98.11	95.21
	MV-Arc-Softmax-a (0.3)	99.76	99.33	99.30	95.50	95.46	89.41	98.06	95.45
	MV-AM-Softmax-f (0.25)	99.79	99.33	**99.31**	95.74	95.45	**89.69**	98.05	**95.70**
	MV-AM-Softmax-a (0.2)	**99.79**	99.33	99.30	**96.27**	**95.63**	89.19	98.00	95.30

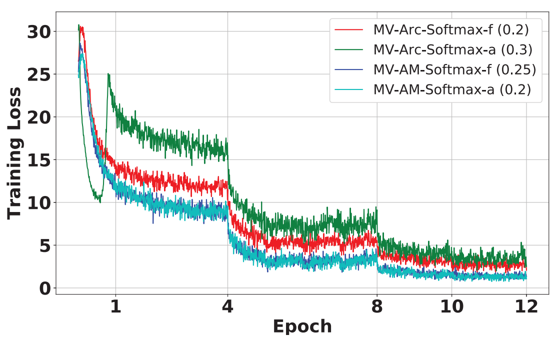

Figure 2: Convergence of MV-Softmax. From the curves, we can see that our MV-Softmax loss functions have a good behavior of convergence.

method has a good behavior of convergence.

Results on LFW, CALFW, CPLFW, AgeDB, CFP

Table 3 provides the quantitative results of all the competitors on LFW, CALFW, CPLFW, AgeDB and CFP. The bold number in each column represents the best result. For the LFW accuracy and its BLUFR protocol with different false alarm rates (*e.g.*, 1e-3, 1e-4, 1e-5), it is well-known that these protocols are typical and easy for face recognition. For instance, at LFW accuracy and TPR@FAR=1e-3 and 1e-4, almost all the competitors can achieve 99% performance. So the improvement of our MV-Softmax losses is not quite large. For the BLUFR with TPR@FAR=1e-5, we can see that the naive fusion HM-AM-Softmax outperforms the baseline Softmax, the simple mining-based losses and the margin-based ones. Despite this, our MV-AM-Softmax still achieves about 0.2% improvement. On CALFW, CPLFW, AgeDB and CFP test sets, we also observe that our MV-Softmax losses are better than the state-of-the-art alternatives in most of cases. Nevertheless, we can see that the improvements of our method in these test sets are not by a large margin. The reason is that the test protocol is relatively easy and the performance of all the methods on these test sets are near saturation. So there is an urgent need to test the performance of all the competitors on new test sets or test with more complicated protocols.

Table 4: Verification performance (%) of different loss functions on the test set RFW.

Method	RFW			
	Caucasian	Indian	Asian	African
Softmax	98.33	93.33	93.16	91.33
F-Softmax	97.50	90.30	91.16	88.33
HM-Softmax	98.66	93.49	92.83	90.50
A-Softmax	98.83	94.33	93.33	91.33
Arc-Softmax	98.83	96.16	93.66	95.00
AM-Softmax	99.16	96.16	94.46	95.83
F-Arc-Softmax	98.99	95.83	94.16	95.50
F-AM-Softmax	99.16	96.66	93.66	95.00
HM-Arc-Softmax	98.66	94.33	94.16	96.66
HM-AM-Softmax	99.16	94.66	93.33	96.00
MV-Arc-Softmax-f	98.66	**96.83**	94.50	96.50
MV-Arc-Softmax-a	98.00	94.66	94.83	95.99
MV-AM-Softmax-f	99.00	94.99	94.83	**96.66**
MV-AM-Softmax-a	**99.33**	95.83	**95.66**	95.83

Results on RFW

Firstly, we evaluate all the competitors on the recent proposed new test set RFW (Wang et al. 2018d). RFW is a face recognition benchmark for measuring racial bias, which consists of four test subsets, namely Caucasian, Indian, Asian and African. Tables 4 displays the performance comparison of all the involved methods. From the values, we can conclude that the results on the four subsets exhibit the same trends, *i.e.*, the margin-based losses are better than the baseline Softmax loss and the mining-based losses. The improvement by simply combining the margin-based and mining-based losses is limited. Our mis-classified guided ones, which explicitly emphasize on the mis-classified feature vectors for training, are more consistent with the dis-

Table 5: Performance (%) of different loss functions on MegaFace and Trillion-Pairs Challenge.

Method	MegaFace		Trillion-Pairs	
	Id.	Veri.	Id.	Veri.
Softmax	93.94	94.76	60.06	59.00
F-Softmax	91.60	93.06	51.14	48.32
HM-Softmax	93.95	95.53	61.34	60.07
A-Softmax	94.18	95.26	60.34	59.01
Arc-Softmax	97.28	97.58	70.80	68.12
AM-Softmax	97.69	97.82	74.00	71.57
F-Arc-Softmax	97.51	97.81	70.65	69.06
F-AM-Softmax	95.75	97.75	73.82	72.18
HM-Arc-Softmax	97.43	97.56	70.08	68.16
HM-AM-Softmax	97.48	97.64	73.89	71.63
MV-Arc-Softmax-f	97.52	98.01	73.90	71.28
MV-Arc-Softmax-a	97.74	97.62	75.44	74.69
MV-AM-Softmax-f	97.95	97.85	75.92	74.45
MV-AM-Softmax-a	**98.00**	**98.31**	**76.94**	**75.93**

criminative feature learning. Therefore, they inherently absorb the merits of feature margin and feature mining into a unified loss function. They usually achieve more discriminative face features and can get higher performance than previous alternatives.

Results on MegaFace and Trillion-Pairs

We then test all the competitors with more complicated protocols. Specifically, the identification (Id.) Rank-1 and the verification (Veri.) TPR@FAR=1e-6 on MegaFace, the identification (Id.) TPR@FAR=1e-3 and the verification (Veri.) TPR@FAR=1e-9 on Trillion-Pairs are reported in Table 5. From the numbers, we can observe that our MV-AM-Softmax-a achieves the best performance over the baseline Softmax loss, the mining-based Softmax losses, the margin-based softmax losses and the naive combinations of mining-based and margin-based losses, on both MegaFace and Trillion-Pairs Challenge. Specifically, on MegaFace, for our proposed MV-AM-Softmax-a, it obviously beats the best margin-based competitor AM-Softmax loss by a large margin (about 0.3% on identification and 0.5% on verification). Compared with the naive fusions of mining-based and margin-based losses, our improved MV-AM-Softmax-a loss is also better than them. Moreover, compared the MV-Softmax-a with MV-Softmax-f, we can say that the adaptive re-weighted function Eq. (8) is generally better than the fixed one Eq. (7). This is reasonable because for more difficult mis-classfied feature vectors, they should be more important for discriminative feature learning. In Figure 3, we also draw both of the CMC curves to evaluate the performance of face identification and the ROC curves to evaluate the performance of face verification on MegaFace Set 1. From the curves, we can see the similar trends at other measures. On Trillion-Pairs Challenge, we can observe that the results exhibit the same trends that emerged on MegaFace test set. Besides, the trends are more obvious. In particular, we achieve at least 3% improvements at both the identification and the verification on Trillion-Pairs Challenge. In this experiment, we have clearly demonstrated that our MV-AM-Softmax-a approach is superior for both the identification and verification tasks, especially when the false positive rate is very low. To sum up, by inheriting the advantages of both margin-based and mining-based Softmax losses, our new desined mis-classified guided one has shown its strong generalization ability for face recognition.

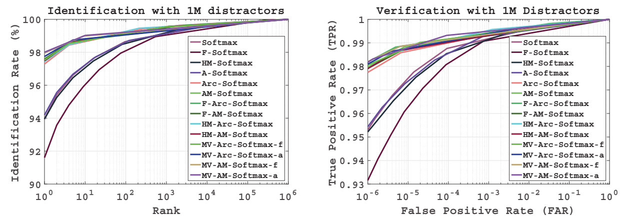

Figure 3: From Left to Right: CMC curves and ROC curves of different loss functions with 1M distractors on MegaFace Set 1.

Conclusion

This paper has proposed a simple yet very effective loss function, namely mis-classified vector guided softmax loss (*i.e.*, MV-Softmax), for the task of face recognition. In specific, MV-Softmax loss explicitly concentrates on optimizing the mis-classified feature vectors. Thus it semantically inherits the motivations of feature margin and feature mining into a unified loss function. Consequently, it exhibits a higher performance than the baseline Softmax loss, the current mining-based losses, margin-based losses and their naive fusions. Extensive experiments on several face recognition benchmarks have validated the effectiveness of our new approach over the state-of-the-art alternatives.

References

Deepglint. 2018. http://trillionpairs.deepglint.com/overview.

Deng, J.; Guo, J.; Xue, N.; and Zafeiriou, S. 2019. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*.

Feng, Z.-H.; Kittler, J.; Awais, M.; Huber, P.; and Wu, X.-J. 2017. Wing loss for robust facial landmark localisation with convolutional neural networks. *arXiv:1711.06753*.

Guo, Y.; Zhang, L.; Hu, Y.; He, X.; and Gao, J. 2016. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *ECCV*.

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

Hu, G.; Yang, Y.; Yi, D.; Kittler, J.; Christmas, W.; Li, S. Z.; and Hospedales, T. 2015. When face recognition meets with deep learning: an evaluation of convolutional neural networks for face recognition. In *CVPRW*.

Hu, G.; Peng, X.; Yang, Y.; Hospedales, T. M.; and Verbeek, J. 2017. Frankenstein: Learning deep face representations using small data. *TIP*.

Huang, G.; Ramesh, M.; and Miller., E. 2007. Labeled faces in the wild: A database for studying face recognition in unconstrained enviroments. *Technical Report*.

Kemelmacher-Shlizerman, I.; Seitz, S. M.; Miller, D.; and Brossard, E. 2016. The megaface benchmark: 1 million faces for recognition at scale. In *CVPR*.

Liang, X.; Wang, X.; Lei, Z.; Liao, S.; and Li., S. 2017. Soft-margin softmax for deep classification. In *ICONIP*.

Liao, S.; Lei, Z.; Yi, D.; and Li, S. Z. 2014. A benchmark study of large-scale unconstrained face recognition. In *ICB*.

Lin, Y.; Goyal, P.; and Girshick., R. 2017. Focal loss for dense object detection. In *ICCV*.

Liu, W.; Wen, Y.; Yu, Z.; Li, M.; and Song., L. 2017. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*.

Liu, H.; Zhu, X.; Lei, Z.; and Li, S. Z. 2019a. Adaptiveface: Adaptive margin and sampling for face recognition. In *CVPR*.

Liu, Y.; Shi, H.; Si, Y.; Shen, H.; Wang, X.; and Mei, T. 2019b. A high-efficiency framework for constructing large-scale face parsing benchmark. *arXiv preprint arXiv:1905.04830*.

Liu, Z.; Hu, G.; and Wang, J. 2019. Learning discriminative and complementary patches for face recognition. In *FG*.

Liu, W.; Wen, Y.; and Yu, Z. 2016. Large-margin softmax loss for convolutional neural networks. In *ICML*.

Moschoglou, S.; Papaioannou, A.; Sagonas, C.; Deng, J.; Kotsia, I.; and Zafeiriou, S. 2017. Agedb: the first manually collected, in-the-wild age database. In *CVPRW*.

Nech, A., and Kemelmacher, I. 2017. Level playing field for million scale face recognition. In *CVPR*.

Ranjan, R.; Castillo, C.; and Chellappa., R. 2017. L2-constrained softmax loss for discriminative face verification. *arXiv preprint arXiv:1703.09507*.

Schroff, F.; Kalenichenko, D.; and Philbin., J. 2015. Facenet: A unified embedding for face recognition and clustering. In *CVPR*.

Sengupta, S.; Chen, J.-C.; Castillo, C.; Patel, V. M.; Chellappa, R.; and Jacobs, D. W. 2016. Frontal to profile face verification in the wild. In *WACV*.

Shi, H.; Wang, X.; Yi, D.; Lei, Z.; Zhu, X.; and Li, S. Z. 2017. Cross-modality face recognition via heterogeneous joint bayesian. *SPL*.

Shrivastava, A.; Gupta, A.; and Girshick., R. 2016. Training region-based object detectors with online hard example mining. In *CVPR*.

Simonyan, K., and Andrew, Z. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sun, Y.; Wang, X.; and Tang., X. 2014. Deep learning face representation from predicting 10,000 classes. In *CVPR*.

Sun, Y.; Wang, X.; and Tang., X. 2015. Deeply learned face representations are sparse, selective, and robust. In *CVPR*.

Taigman, Y.; Yang, M.; and Ranzato., M. 2014. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*.

Wang, F.; Xiang, X.; Chen, J.; and Yuille., A. 2017a. Normface: l_2 hypersphere embedding for face verification.. In *ACM MM*.

Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; and Tang, X. 2017b. Residual attention network for image classification. *arXiv:1704.06904*.

Wang, F.; Chen, L.; Li, C.; Huang, S.; Chen, Y.; Qian, C.; and Loy, C. C. 2018a. The devil of face recognition is in the noise. In *ECCV*.

Wang, F.; Cheng, J.; Liu, W.; and Liu, H. 2018b. Additive margin softmax for face verification. *SPL*.

Wang, H.; Wang, Y.; Zhou, Z.; Ji, X.; Li, Z.; Gong, D.; Zhou, J.; and Liu, W. 2018c. Cosface: Large margin cosine loss for deep face recognition. *arXiv preprint arXiv:1801.09414*.

Wang, M.; Deng, W.; Hu, J.; Peng, J.; Tao, X.; and Huang, Y. 2018d. Racial faces in-the-wild: Reducing racial bias by deep unsupervised domain adaptation. *arXiv:1812.00194*.

Wang, X.; Wang, S.; Zhang, S.; Fu, T.; and Mei, T. 2018e. Support vector guided softmax loss for face recognition. *arXiv preprint arXiv:1812.11317*.

Wang, X.; Zhang, S.; Lei, Z.; Liu, S.; Guo, X.; and Li, S. Z. 2018f. Ensemble soft-margin softmax loss for image classification. *arXiv preprint arXiv:1805.03922*.

Wang, X.; Wang, S.; Wang, J.; Shi, H.; and Mei, T. 2019. Co-mining: Deep face recognition with noisy labels. In *ICCV*.

Wang, X.; Guo, X.; and Li, S. Z. 2015. Adaptively unified semi-supervised dictionary learning with active points. In *ICCV*.

Wang, J.; Zhou, F.; and Wen., S. 2017. Deep metric learning with angular loss. In *ICCV*.

Wen, Y.; Zhang, K.; and Li, Z. 2016. A discriminative feature learning approach for deep face recognition. In *ECCV*.

Zhang, S.; Zhu, X.; Lei, Z.; Shi, H.; Wang, X.; and Li, S. Z. 2017. Faceboxes: A cpu real-time face detector with high accuracy. In *IJCB*.

Zhang, S.; Wang, X.; Lei, Z.; and Li, S. Z. 2019. Faceboxes: A cpu real-time and accurate unconstrained face detector. *Neurocomputing*.

Zheng, T.; Deng, W.; Hu, J.; and Hu, J. 2017. Cross-age lfw: A database for studying cross-age face recognition in unconstrained environments. *arXiv:1708.08197*.

Zheng, T.; Deng, W.; Zheng, T.; and Deng, W. 2018. Cross-pose lfw: A database for studying crosspose face recognition in unconstrained environments. *Tech. Rep*.

Zheng, Y.; Pal, D. K.; and Savvides, M. 2018. Ring loss: Convex feature normalization for face recognition. In *CVPR*.