AdaDistill: Adaptive Knowledge Distillation for Deep Face Recognition

Fadi Boutros[1], Vitomir Štruc[2], and Naser Damer[1,3]

[1] Fraunhofer Institute for Computer Graphics Research IGD, Darmstadt, Germany
[2] University of Ljubljana, Ljubljana, Slovenia
[3] Department of Computer Science, TU Darmstadt, Darmstadt, Germany
{fadi.boutros,naser.damer}@igd.fraunhofer.de,vitomir.struc@fe.uni-lj.si

Abstract. Knowledge distillation (KD) aims at improving the performance of a compact student model by distilling the knowledge from a high-performing teacher model. In this paper, we present an adaptive KD approach, namely AdaDistill, for deep face recognition. The proposed AdaDistill embeds the KD concept into the softmax loss by training the student using a margin penalty softmax loss with distilled class centers from the teacher. Being aware of the relatively low capacity of the compact student model, we propose to distill less complex knowledge at an early stage of training and more complex one at a later stage of training. This relative adjustment of the distilled knowledge is controlled by the progression of the learning capability of the student over the training iterations without the need to tune any hyper-parameters. Extensive experiments and ablation studies show that AdaDistill can enhance the discriminative learning capability of the student and demonstrate superiority over various state-of-the-art competitors on several challenging benchmarks, such as IJB-B, IJB-C, and ICCV2021-MFR [4].

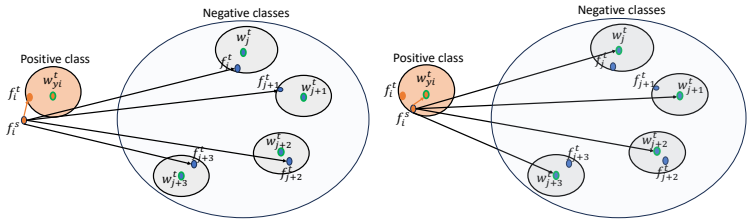
Keywords: Face recognition · Computer vision · Knowledge distillation

1 Introduction

State-of-the-art (SOTA) face recognition (FR) models [14, 23, 41] rely on training deep neural networks (DNN) [20] with millions of trainable parameters and high computational cost. However, deploying such models on mobile and edge devices remains a challenge due to the limited computational capacities of these devices [15]. Designing efficient network architectures [2, 9, 44] and compressing existing deep neural networks (DNNs) [4], using e.g. parameter pruning or model quantization, are common approaches to mitigate this challenge [8]. While these approaches have been shown to reduce the required computational resources [4,9,44], they commonly lead to lower verification accuracies compared to the more computationally expensive SOTA models [15].

Knowledge distillation is an effective technique to improve the performance of compact models by transferring the knowledge from a large and well-performing

[4] https://github.com/fdbtrs/AdaDistill



(a) AdaDistill - Early stage **(b)** AdaDistill - Later stage

Fig. 1: Overview of our adaptive KD approach. At an early stage of training, f_i^s (from the student) is pushed to be close to the counterpart f_i^t obtained from the teacher and far from other f_j^t of different classes. As training goes on, f_i^s is pushed to be close to its class center, w_{yi}^t (distilled from the teacher) and far from all other class centers, w_j^t.

teacher model to a comparably smaller student model [17,21,24,27]. With guidance from the teacher, the student can learn more effectively and achieve high accuracy, despite having fewer parameters. The idea of KD was originally proposed by Bucilua et al. [7] in 2006 and popularized in 2015 by Hinton et al. [21]. While it was initalliy applied to image classification [21], it was later widely adopted and advanced for different machine-learning tasks. In FR, for example, several works proposed to improve compact FR performances using KD [16,17,24,27,39]. From such works, [17] opted to adapt and modify the traditional triplet loss in the distillation process to include a dynamic margin. Others opted to adopt a rank-based loss to optimize the distillation process based on pair verification performance [24]. The feature direction and sample distributions transfer to the student through an angular loss was also recently proposed [16]. While [27] proposed a feature-based KD approach with reverse distillation and a student proxy, this required multiple phases of student-teacher training and imposed strict constraints on the student to imitate the teacher embeddings. On the other hand, [39] fixed class centers learned by the teacher model and trained the student with a margin-based softmax loss. However, because these classes are fixed, this might not be optimal for all the training stages.

In this study, we introduce a novel adaptive knowledge distillation approach termed AdaDistill. AdaDistill addresses several key challenges of previous approaches by eliminating the reliance on fixed class centers, avoiding the need for recursive training, and circumventing the need for multiple loss balance. Furthermore, it accounts for the student-teacher gap, ensuring effective knowledge transfer from the teacher to its student in an adaptive manner. We embedded our adaptive distillation concept into the margin penalty softmax loss [14], aiming at maximizing the distance between features of different classes by incorporating a margin penalty between feature representations of each sample and its class center, which is distilled from the teacher. Additionally, we consider the capacity gap, in terms of learning capabilities, between the shallow student and large teacher models and propose to transfer less complex knowledge at an early stage of training and more complex one at a later stage of training.

To achieve that, an adaptive procedure based on the exponential moving average over class-conditioned feature representations is introduced that allows for the distilling of knowledge at the early stages based on sample-to-sample

comparisons, while at the later stages, more complex information captured by sample-to-class-center comparisons is exploited, as illustrated in Fig. 1b. This comparison process evolves dynamically over the training stages and is determined by the student based on its ability to imitate feature representations from the teacher, i.e., using positive cosine similarity between feature representations from the teacher and the student of each sample. We also incorporate the importance of hard samples in the class center estimation, as induced by the teacher. The proposed AdaDistill does not require manually tuning additional hyperparameters other than the margin penalty value of the softmax loss, where class centers estimation and the importance of hard samples are dynamically adaptive over the training iteration. We present several ablation studies to support our AdaDistill design choices. We also conduct extensive evaluations on 10 benchmarks, demonstrating the superiority of our AdaDistill over SOTA KDs.

2 Related Works

KD approaches in the literature can be categorized, based on the form of the distilled knowledge, under three categories, response-based, feature-based, and relation-based KD [18]. Response-based KD refers to mimicking the output prediction between the student and the teacher. The work by Hinton et al. [21] was one of the pioneer works proposing response-based KD that minimizes the Kullback-Leibler (KL) divergence between the soft class probabilities of the teacher and the student. This inspired several research works, proposing KD approaches to transfer a diversity of knowledge forms from the teacher to the student beyond class probabilities. Feature-based KDs, also known as instance-based [27], utilizes instance-wise knowledge from intermediate feature representations as additional supervision to train the student model. Within this category, FitNet [36] proposed to distill the knowledge from the intermediate feature maps by utilizing a regressor to match the dimensions of the intermediate feature maps between the teacher and the student. ShrinkTeaNet [16] presented an angular loss to transfer the feature direction and sample distributions from the teacher embedding space to its student. MarginDistillation [39] trained a student model with margin-based softmax loss and fixed class centers learned by the teacher. TripletDistillation [17] extended the traditional triplet loss used in deep face recognition by incorporating a dynamic margin. This dynamic margin is derived from the similarity structures among different identities from the teacher model. Very recently, Li et al. [27] proposed a feature-based KD approach with reverse distillation and a student proxy, aiming at minimizing the intrinsic gap between the teacher and the student model. Unlike response-based and feature-based KD, which utilizes the output of a single layer or pairs of data samples, relation-based KD consideres the relation between several layers or data points within a batch of samples. Here, DarkRank [10] proposed to distill the relative similarity based on ranking between samples within each training batch. CCKD [35] presented a KD approach to transfer the pairwise similarity on the instance level and the correlation between the instances on the batch level. RKD [33] proposed to transfer relations among data points learned by the teacher to the student model using distance-wise and angle-wise distillation losses. EKD [24] introduced

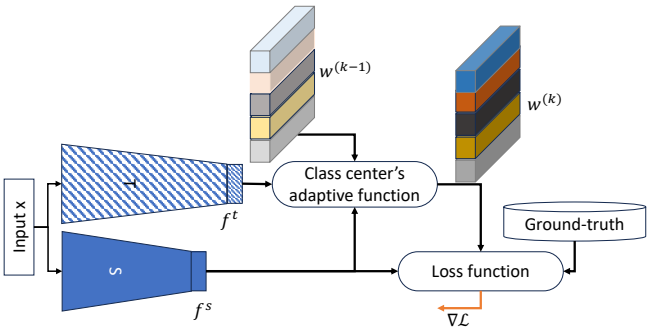


Fig. 2: Overview of AdaDistill. A batch of samples x from the $k-th$ training iteration is passed to T and S to output two sets of feature representations f^t and f^s, respectively. Then, the class centers $w^{(k)}$ are calculated (Eq. 6) based on the importance of hard samples and the learning capability of the student (Eq.8). Finally, the loss is calculated (Eq. 4) and the weights of S are updated.

a rank-based loss, aiming at selecting key pair-relations based on face verification metrics to be distilled to the student model.

Our proposed AdaDistill aims at distilling knowledge from a teacher to a student model by leveraging their respective feature representations from the embedding space. Thus, it is a feature-based method. Compared to ReFo [27], which aims at training a student model to imitate exact teacher embedding space, our AdaDistill imposes less strict constraints on the student by pushing the samples to be relatively closer to their class center than other negative class centers. Also, unlike ReFo, our AdaDistill does not require s student proxy or reverse distillation through multiple phases of student-teacher training. At an early stage of student training, our AdaDistill and ReFo [27] share similar learning objectives, while at the later stage of training, our AdaDistill and MarginDistillation [39] share similar learning objectives. Different from MarginDistillation [39] that utilized fixed class centers from the teacher, our AdaDistill adapts the class center at different stages of the model training, enabling better convergence and higher verification accuracies. Compared to ShrinkTeaNet [16] that minimized the angular distance between the teacher and student embedding representations for each sample, our AdaDistill transfers richer and more discriminative information from class centers learned by the teacher to its student.

Similarly to [27, 39], and differently from [16, 24, 33, 35, 36], our AdaDistill utilizes only the distillation loss as the main training objective for the student without additional supervision from the classification loss. This mitigates the challenge of simultaneously optimizing different learning objectives or balancing the values of interest (i.e., loss weighting terms).

3 Methodology

This section first presents preliminaries on conventional feature-based KD. Next, it presents the concept of template distillation through the Softmax loss. Finally, it introduces our AdaDistill, which considers the gap between the student and the teacher, as well as the importance of hard samples over the distillation training. An overview of the proposed AdaDistill approach is presented in Fig. 2.

3.1 Preliminary: Knowledge distillation

Throughout the paper, we denote the teacher and student models as T and S, respectively. The training dataset is denoted as D and consists of training samples $x \in X$ and their corresponding class labels $y \in \{1, \ldots, c\}$, where c is the number of classes. The feature representations of sample x_i, extracted by T and S, are denoted as f_i^t and f_i^s, respectively.

Given a pretrained teacher T, the general loss function for training S in the KD paradigm can generally be formulated as:

$$\mathcal{L}_S = \lambda \mathcal{L}_{main} + \beta \mathcal{L}_{KD}, \tag{1}$$

where $\mathcal{L}_{main}$ is the task-specific loss function, e.g. Cross-entropy for classification learning, and $\mathcal{L}_{KD}$ is the KD loss (e.g., KL divergence) used to transfer the knowledge from T to S. λ and β are weightings parameters. Eq. 1 represents a common loss for KD and λ and β do not necessarily sum up to one. Approaches, such as [27], set λ to zero, while others utilize both losses [24]. Response-based KD approaches, such as [21], utilize the KL divergence to match the output distributions, i.e., the classification output, of the teacher and student models. However, in FR, matching the final output is less effective, as the goal of FR is learning to produce discriminative feature representations, instead of only accurate classification with the Softmax layer [24,27]. Therefore, SOTA feature-based KD [27] aim to mimic intermediate feature representations, f_i^t and f_i^s, instead of the final output probabilities. We follow these insights with AdaDistill, and, therefore, do not consider the KD term in the distillation process, i.e., $\beta = 0$.

Several previous works [27, 44] proposed to replace KL by the Euclidean distance, i.e. mean squared error (MSE) for feature-based distillation. MSE is widely adopted in the recent SOTA approaches [27] and is given by:

$$\mathcal{L}_{mse} = \frac{1}{N} \sum_{i \in n} \|f_i^s - f_i^t\|_2^2, \tag{2}$$

where N is the batch size. Feature distillation with MSE proves to improve S model verification accuracies [27]. However, this approach faces two main challenges. First, it assumes that the compact student model can learn to imitate the teacher embedding spaces using a shallow network architecture. However, achieving this goal is challenging due to the large gap, in terms of network architecture size between the compact S and the large T, as well as the limited learning capability of the shallow S network [27, 44]. Several previous works proposed to address this challenge using, for example, multiple phases of student and teacher training [27, 44] or by using teacher assistants [30]. which is computationally expensive, especially during the training phase. Second, as feature distillation with MSE aims at imitating the exact teacher embedding space [27,44] (which is challenging due to the capacity gap between S and T), it prevents the student model from establishing its own embedding space or explicitly learning the proximity between samples and their class centers and negative class centers. We therefore propose in this work an adaptive feature-based KD approach that addresses these challenges by 1) Instead of imitating the exact teacher embedding space,

we train the student to learn to push samples to be relatively close to their class centers and far from other class centers, and 2) we further propose an adaptive approach that guided the student model to learn less complex knowledge at an early stage of training and more complex knowledge at a later stage of training.

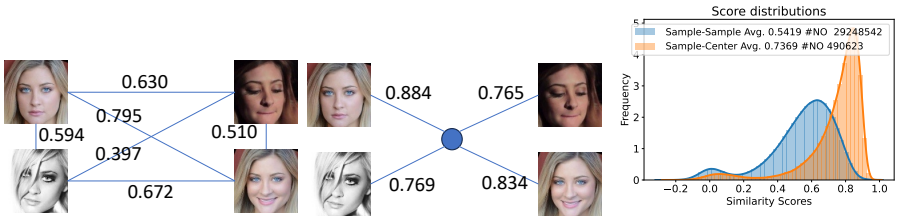
3.2 Feature distillation through classification loss

Preliminary on classification loss function: The Softmax loss and its variants, i.e. margin penalty softmax losses, are commonly used loss functions to train FR models [3, 14, 28, 41] that implicitly learn to produce discriminative feature representations through learning a multiclass classification. The Softmax loss refers to applying cross-entropy between the output of the softmax activation function and ground-truth labels. Margin penalty softmax losses extend the softmax loss by introducing a margin penalty on the cosine angle (or its cosine) of the feature representation of sample i and its class center. This aims at pushing the decision boundary of the softmax, in turn enhancing intra-class compactness and inter-class discrepancy. In this work, we utilize two versions of the margin penalty softmax loss, ArcFace [14] and CosFace [41]. It should be noted that our proposed approach can be deployed using any margin penalty softmax loss. We opt for ArcFace and CosFace, as examples, as they are widely used classification losses in the FR literature [14, 41]. Formally, the Angular Margin Penalty-based Losses (AML), ArcFace and CosFace, are given by:

$$\mathcal{L}_{AML} = \frac{1}{N}\sum_{i \in N} -log\frac{e^{s(cos(\theta_{y_i}+m1)-m2)}}{e^{s(cos(\theta_{y_i}+m1)-m2)} + \sum\limits_{j=1, j\neq y_i}^{c} e^{s(cos(\theta_j))}}, \tag{3}$$

where N is the batch size, y_i is the class label of sample i (in range $[1, c]$, where c is the number of classes), θ_{y_i} is the angle between the features f_i and the y_i-th class center w_{y_i}. $f_i \in \mathbb{R}^d$ is the deep feature embedding of the last fully connected layer of size d. $w_{y_i} \in \mathbb{R}^d$ is the y_i-th column of weights $W \in \mathbb{R}^{d \times c}$ of the classification layer. θ_{y_i} is defined as $f_i w_{y_i}^T = \|f_i\| \|w_{y_i}\| cos(\theta_{y_i})$ [28]. The weights and the feature norms are fixed to $\|w_{y_i}\| = 1$ and $\|f_i\| = 1$, respectively, using l_2 normalization, as defined in [28, 41]. The decision boundary, in this case, depends on the cosine of the angle between f_i and w_{y_i}. In ArcFace [14], $m1 > 0$ and $m2 = 0$, where $m1$ is an additive angular margin penalty, aiming at enhancing the intra-class compactness and inter-class discrepancy. In CosFace [41], $m1 = 0$ and $m2 > 0$ is additive cosine margin penalty. Lastly, s is the scaling parameter [41]. The decision boundary of AML is $cos(\theta_{y_i} + m1) - m2 - cos(\theta_j) = 0$. The model, in this case, learns to optimize f_i and w_j as $cos(\theta_j) \leq cos(\theta_{y_i} + m1) - m2$, i.e., pushing the sample f_i to be closer to its class center than other negative class centers. In the backpropagation phase, the weights of both f_i and w_j are updated incrementally after each iteration using the Gradient Decent optimization algorithm [37]. In the following sections, $\mathcal{L}_{AML}$ will be denoted as $\mathcal{L}_{Arc}$ and $\mathcal{L}_{Cos}$ for ArcFace and CosFace, respectively.

AMLDistill: Additive margin penalty distillation: The class centers w_j^t learned by the teacher model contain more identity-representative information



(a) Sample-Sample **(b)** Sample-Center **(c)** Score distributions

Fig. 3: Sample-sample and Sample-center similarity score distributions. Fig. 3a and 3b presented an example of matching samples with others (of the same identity) and samples with their class center. Fig. 3c matching samples with samples achieved lower similarity scores than matching samples with their class centers.

than f_i^t of one sample. Fig. 3c presents two similarity score distributions obtained by comparing samples with all other samples of each identity (blue plot) and samples with their class centers (orange plot). It can be noticed that samples can match with their class center with a higher similarity score than matching sample-to-sample of the same identity, indicating that class centers better represent the identity classes than any single sample. For the visualization, the similarity scores are calculated using the cosine similarity between embeddings [5] from CASIA-WebFace (0.5M images). Thus, we train S, similarly to [39], to optimize the angular distance between f_i^s (from S) and w_j^t (from T). Specifically, we propose to train the student model to push f_i^s in a way that is closer to its class center w_{yi}^t and farther from other negative class centers w_j^t. Specifically, we set $w_j^s = w_j^t$. The modified loss function is given by:

$$\mathcal{L}_{AMLDistill} = \frac{1}{N}\sum_{i\in N} -log\frac{e^{s(cos(\theta^t_{y_i}+m1)-m2)}}{e^{s(cos(\theta^t_{y_i}+m1)-m2)} + \sum\limits_{j=1,j\neq y_i}^{C} e^{s(cos(\theta_j^t))}}, \tag{4}$$

where $\theta^t_{y_i}$ is the angle between the features f_i^s and the y_i-th class center $w^t_{y_i}$ learned by the teacher model. $cos(\theta^t_{y_i})$, in this case, is defined as follow:

$$cos(\theta^t_{y_i}) = f_i^s (w^t_{y_i})^T, \tag{5}$$

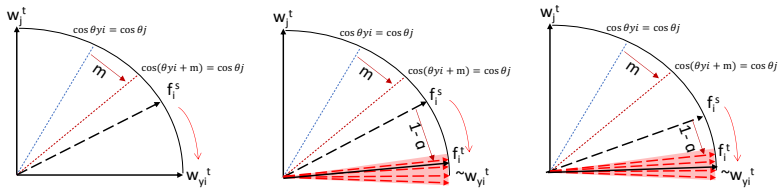
where f_i^s is the feature representation learned by the student and w_{yi}^t is its class.

In the backpropagation phase, the weights w_j^t are fixed and only f_i^s are updated after each epoch using the Gradient Decent optimization algorithm [37]. Unlike previous KD approaches [16,24,27,33,35,36], AMLDistill (Eq. 4) requires access only to the precomputed weight of the last layer of T (i.e., $w_j^s = w_j^t$) and does not require passing the training samples into the teacher model. Our models are denoted as ArcDistill, when the utilized margin penalty is $m1$ (ArcFace) and as CosDistill when the utilized margin penalty is $m2$ (CosFace).

3.3 AdaDistill: Adaptive AMLDistill

Given that w_{yi}^t captures more complex and identity-representative information than f_i^t and considering the substantial discrepancy between T and S model, S

[5] We utilized ResNet100 [20] trained using ArcFace loss [14] on MS1MV2 dataset [14, 19] to obtain the embedding



(a) ArcDistill **(b)** AdaDistill (Early-stage) **(c)** AdaDistill (Later-stage)

Fig. 4: Different distillation approaches using fixed class center (Fig. 4a) and adaptive class center (Figs. 4b and 4c). In ArcDistill (Figrue 4a), f_i^s is pushed with penalty margin to fixed class center w_{yi}^t. Figs. 4b and 4c illustrate the adaptive estimation of class center w_{yi}^t using exponential moving average (Eq. 6). At an early stage of training (Fig. 4b), the α value (positive cosine similarity between f_i^s and f_i^t) is small and the class center is close to f_i^t. As training progresses (Fig. 4c), α increases and the new class center is estimated to be close to the average of all f_i^t of class y_i.

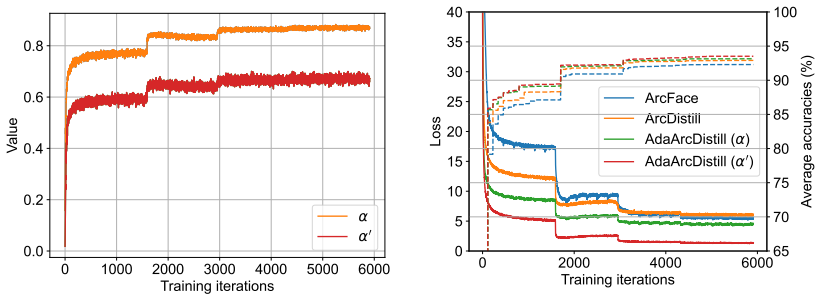
is not fully capable of mimicking the distance between f_i^s and the teacher-centers w_{yi}^t. This can be observed from Fig. 5b, which presents the training losses over the training iterations of various models trained with different loss functions. It can be clearly noticed that the model trained with ArcDistill (orange plot), i.e., S aimed at minimizing the distance between f_i^s and the teacher-centers w_{yi}^t, suffers from low convergence in comparison to AdaArcDistill, which we introduce in the remainder of this section. To address the aforementioned challenges, we propose an approach based on Exponential Moving Average (EMA) to refine w_{yi}^t over the model training stage. Specifically, given that the average of all f_i^t approximately corresponds to the $i-th$ class center, we propose to estimate w_{yi} using f_i^t in different training stages. Let θ^{easy} and θ^{hard} be the angle between f_i^s and f_i^t and f_i^s and w_{yi}, respectively. The key idea is that the student at an early stage of training can more easily learn f_i^s as $cos(\theta_j^{easy}) \leq cos(\theta_{y_i}^{easy} + m)$ than $cos(\theta_j^{hard}) \leq cos(\theta_{y_i}^{hard} + m)$. As the training progresses, S will be more capable of learning complex patterns from the identity center. Formally, w_{yi} is updated after each $k-th$ training iteration as:

$$w_{yi}^{(k)} = \alpha w_{yi}^{(k-1)} + (1-\alpha)(f_i^{t(k)})^T, \tag{6}$$

where k is the current training steps. T is the transpose function. $\alpha \in [0,1]$ is the momentum parameter and set to the positive cosine similarity between f_i^s and f_i^t from the $k-th$ batch, given by:

$$\alpha = \lfloor Cos(f_i, f_i^t) \rceil_0^1, \tag{7}$$

where $\lfloor . \rceil$ refers to clipping the value between 0 and 1. Based on the refined w, the student model is trained as in Eq. 4. At an early stage of training, $\alpha \approx 0$ and thus, $cos(\theta_{y_i}^t) = f_i^s (w_{yi})^T, w_{yi} \approx f_i^t$. The student, in this case, learns to minimize the distance between f_i^s and its counterpart f_i^t. This early training stage is referred to as an easy stage, where S learns less complex information. As seen in Fig. 5a, the α value is increased over the training iterations. At a later stage of training $\alpha \approx 1$ and thus, $cos(\theta_{y_i}^t) = f_i^s (w_{yi})^T, w_{yi} \approx Avg(f_{yi}^t)$, i.e, $w_{y_i}^t$ is the average embedding of f^t belong to class y_i. In this later stage of training,



(a) **(b)**

Fig. 5: Fig. 5a (Left): Average α and α' values over the training iterations as defined in Eq.7 and 8, respectively. Fig. 5b (Right): Loss values and average verification accuracies over training iterations of students trained with different loss functions. The losses are in solid lines and the average accuracies are in dashed. The average accuracies are calculated on five benchmarks, LFW [22], CFP-FP [38], AgeDB-30 [31], CA-LFW [48] and CP-LFW [47], described in Section 4.1. AdaDistill (α) and AdaDistill (α') facilitated better convergence and achieved higher accuracies in comparison to the ArcDistill and standalone ArcFace.

S learns complex information about identity centers. This adaptive distillation function at the early stage and the late stage is simplified visually, along with the function when the class center is fixed, in Fig. 4.

Importance of hard mining: The α, defined in Eq. 7, represents the ability of the student model to mimic the teacher feature space, and thus, the student fully controls α. In this case, there is no consideration of the importance of the hard samples as induced by the teacher model. For that, we propose to weight the α based on the importance of the hard training sample. Specifically, we estimate the difficulty of each training sample in the current batch based on the positive cosine similarity between feature representation f_i^t of sample i (obtained from the teacher) and its estimated class center w_{yi}. The weighted α' is given by:

$$\alpha' = \lfloor Cos(f_i^s, f_i^t) \times Cos((w_{yi}^{(k-1)})^T, f_i^t) \rceil_0^1. \tag{8}$$

In each training iteration, when the value of the weighting term $Cos((w_{yi}^{(k-1)})^T, f_i^t)$ is small, i.e, hard sample, α (Eq. 6) is small and thus $w_{yi}^{(k)}$ is refined toward f_i^t of hard sample. Conversely, when $Cos((w_{yi}^{(k-1)})^T, f_i^t)$ is large, the center $w_{yi}^{(k)}$ is refined toward the average embeddings. The average value of α' over training iteration is shown in Fig. 5a. Our AdaDistill is used to train S with loss defined 4 using refined w_{yi} (Eq 6) and weighted α' (Eq .8). All our models are trained with only the AdaArcDistill, and thus, it mitigates the need for optimizing different learning objectives or tuning loss weighting terms. Our models are noted as AdaArcDistill when the utilized margin penalty is $m1$ (ArcFace) and as AdaCosDistill when the utilized margin penalty is $m2$ (CosFace).

4 Experimental settings

4.1 Datasets

Training: We use MS1MV2 [14, 19] to train our proposed models for fair comparisons with SOTA KD approaches [24, 27]. MS1MV2 is a refined version of

MS-Celeb-1M [19] by [14], containing 5.8M images of 85K identities. In our ablation study, we use also Casia-WebFace [45] and IDiff-Face [5] in evaluating training the student using a different dataset than the one used to train the teacher. Casia-WebFace [45] contains 494,414 images of 10,575 different identities. IDiff-Face [5] is a synthetic dataset and it contains 500k images of 10k identities. Note that all results in this paper are reported using MS1MV2 as a training dataset unless explicitly noted.

Testing: The presented models are evaluated on 10 diverse benchmarks: Labeled Faces in the Wild (LFW) [22], AgeDB-30 [31], Celebrities in Frontal-Profile in the Wild (CFP-FP) [38], Cross-age LFW (CA-LFW) [48], Cross-Pose LFW (CP-LFW) [47], ICCV21-MFR [13], IARPA Janus Benchmark–C [29] (IJB-C) and Benchmark–B [43], MegaFace [25], and the refined MegaFace dataset (MegaFace (R)) [14, 25].

Data preprocessing: Following [14,24,27], all images are aligned and cropped to 112×112, as described in [14], using five landmark points extracted by the Multi-task Cascaded Convolutional Networks (MTCNN) [46]. Also, images are normalized to have pixel values between -1 and 1.

4.2 Evaluation metric

The verification performances on LFW [22], AgeDB-30 [31], CA-LFW [48], CP-LFW [47], and CFP-FP [38] are reported in the form of verification accuracy (%), following their evaluation protocols. We followed the 1:1 mixed verification protocol of IJB-B and IJB-C and report the result as true acceptance rates (TAR) at false acceptance rates (FAR) of 1e-4 and 1e-5 [29, 43]. The ICCV21-MFR [13] contains three test sets: Mask, Children, and Multi-racial (MR-all). Following the ICCV21-MFR evaluation metrics [13], we report the results of the Mask and Children test sets as TAR at FAR=1e-4 and MF-all as TAR at FAR=1e-6. We report the results on MegaFace and MegaFace(R) [14, 25] as Rank-1 identification rate and as TAR at FAR=1e–6 verification accuracy.

4.3 Implementation details

Network architectures: We employ the pretrained ResNet50 (43.59 m parameters and 13.64 GFLOPs) [14, 20] as the teacher. The teacher is trained on MS1MV2 with ArcFace using default training settings (scale parameter s is 64 and the margin m is 0.5) [14]. The weights of the teacher model in all experiments are frozen. The feature embedding size is 512. Following previous SOTA approaches, we use MobileFaceNet [9] (1.19m parameters and 0.45 GFLOPs), noted as MFN, as the student. Our choice for the teacher (ResNet50) and the student (MFN) is to be aligned and to provide a fair comparison with SOTA methods [24, 27, 39]. Additionally, we evaluate our AdaDistill using different pretrained teachers' architectures, ResNet18 (24.02m parameters and 5.23 GFLOPs) [14, 20], ResNet100 (65.15m parameters and 24.21 GFLOPs) [14, 20] and TransFace-B (124.47m parameters and 21.92 GFLOPs) [12].

Student training setups The presented models are implemented using Pytorch [34]. We followed the default setting of ArcFace [14] and CosFace [41] and set the scale parameter s to 64 and margin $m1$ to 0.5 in ArcFace and $m2$

Ablation	Method	LFW	CFP-FP	AgeDB-30	CA-LFW	CP-LFW	Avg.	IJBC 1e-4	IJBC 1e-5
	ResNet50 (Teacher)	99.80	97.63	97.92	96.10	92.43	96.77	96.05	93.96
	MFN (Student)	99.52	91.66	95.82	95.12	87.93	94.01	89.13	81.65
1) KD method	MFN + Vanilla KD (Eq. 2)	99.57	93.77	96.97	95.70	89.01	95.00	91.29	79.79
	MFR + ArcDistill (Eq. 4)	99.63	93.83	96.82	95.48	89.20	94.99	91.70	84.57
	MFN + AdaArcDistill (α) (Eq. 4, 6 and 7)	99.60	94.00	96.88	95.65	89.63	95.15	92.56	88.23
	MFN + AdaArcDistill (α') (Eq. 4, 6 and 8)	99.60	95.04	96.78	95.58	90.08	**95.42**	**92.97**	**89.13**
2) Margin Penalty type and value	MFN + AdaArcDistill (α'), m =0.40	99.53	95.21	96.82	95.54	90.02	95.42	92.98	88.87
	MFN + AdaArcDistill (α'), m =0.45	99.58	95.24	96.78	95.55	90.02	**95.47**	**93.27**	**89.32**
	MFN + AdaArcDistill (α'), m =0.50	99.60	95.04	96.78	95.58	90.08	95.42	92.97	89.13
	MFN + AdaCosDistill (α'), m=30	99.50	94.96	95.95	95.20	90.32	95.19	92.92	**89.26**
	MFN + AdaCosDistill (α'), m=35	99.50	95.07	96.60	95.33	90.02	**95.30**	**93.05**	89.21
	MFN + AdaCosDistill (α'), m=40	99.60	93.98	96.57	95,45	89.35	94.99	93.12	89.09
3) Teacher architecture	ResNet18 (Teacher)	99.67	94.47	97.13	95.70	89.73	95,34	93.99	91.14
	MFN + AdaArcDistill(α'), m=45	99.60	94.30	96.43	95.35	89.13	94,96	92.27	88.10
	ResNet50 (Teacher)	99.80	97.63	97.92	96.10	92.43	96.77	96.05	93.96
	MFN + AdaArcDistill(α'), m=45	99.58	95.24	96.78	95.55	90.02	95.47	93.27	89.32
	ResNet100 (Teacher)	99.83	98.40	98.33	96.13	93.22	97,19	96.39	94.58
	MFN + AdaArcDistill(α'), m=45	99.63	94.26	96.97	95.60	89.28	95,15	92.54	87.48
	TransFace-B (Teacher)	99.85	99.17	98.53	96.20	92.92	97.33	96.55	94.15
	MF+ AdaArcDistill(α'), m=45	99.57	94.23	96.53	95.47	89.87	95.13	92.85	87.98

Table 1: Ablation studies on different KD methods, margin parameters selection, and different teacher architecture. The verification accuracies are reported in (%) on five small benchmarks, LFW, CFP-FP, AgeDB-30, CA-LFW, CP-LFW, and as TAR@FAR of 1e-4 and 1e-5 on large-scale IJB-C benchmark. The results are grouped based on the targeted ablation study. The first two rows present the results of the student and the teacher (used in the 1st and 2nd studies), respectively.

to 0.35 in CosFace(Eq. 4). The selection of the margin values is validated in our ablation study. The batch size in all experiments is set to 512 and the models are trained with a Stochastic Gradient Descent (SGD) optimizer with an initial learning rate of 1e-1. The momentum is set to 0.9 and the weight decays to 5e-4. The learning rate is divided by 10 at 80k, 140k, 210k, and 280k training iterations. During the student training, only random horizontal flipping with a probability of 0.5 is used as data augmentation. The models are trained on 4 Nvidia GeForce RTX 6000 GPUs. Under the same settings, training standalone MFN requires an average of 0.087 seconds to process (forward and backward propagation) a batch of 512 samples. Training MFN with ResNet50 using vanilla KD (Eq. 2) requires an average of 0.207 seconds for a single batch. This time complexity is reduced to 0.134 seconds using ArcDistill. Our AdaArcDistill (α) and AdaArcDistill (α') add a slight burden on training complexity, compared to vanilla KD, with 0.243 and 0.247 seconds, respectively to process a single batch.

5 Results

This section presents rigorous ablation studies to analyze the characteristics of AdaDistill as well as comprehensive comparisons against competing SOTA methods.

5.1 Ablation study

Table 1 presents the verification performances on six benchmarks. We use the average accuracy on the five small benchmarks (LFW, CFP-FP, AgeDB-30, CAL-LFW, and CP-LFW) and the verification accuracy on the large-scale IJB-C as the main criterion for comparing different methods and parameters selection in our ablation and sensitivity studies. The results are reported in (%) using the evaluation metrics described in Section 4.2. The first two rows present the performance of the teacher (ResNet50) and the student (MFN) models, respectively.

Impact of adaptive class centers and importance of hard samples: We first conducted several experiments on MFN using different KD methods, including 1) Vanilla KD (Eq.2), 2) ArcDistill (Eq.4), AdaArcDistill (Eq.4, 6 and 7) and AdaArcDistill (Equations 4, 6 and 8). All distillation methods improved the verification accuracies of MFN. This observation is demonstrated by comparing MFN (second row in Table 1) with the first group of the ablation study (KD method) in Table 1. The model trained with the adaptive class centers outperformed the one trained with fixed class centers. It can be also noticed that the largest improvement is achieved using our AdaArcDistill(α'), emphasizing the benefit of the adaptive class centers and the importance of hard samples. These enhancements in the student performance are also noticed in the model convergence in Fig. 5b, where we plotted the loss values over the training iteration of students trained with different loss functions. Our AdaArcDistill(α') facilitated better convergence in comparison to ArcDistill and the standalone student.

Study on margin penalty: We evaluated the impact of the type and the value of the margin penalty by utilizing two additive margin penalty losses, additive angular margin penalty (AdaArcDistill Eq. 4) and additive cosine margin penalty (AdaCosDistill Eq. 4). For each of the considered losses, we evaluated different penalty values, following their respective works [14,41]. As shown in Table 1, ArcFace with m of 0.45 achieved the best overall performance. For CosFace, the best performance is achieved using m of 0.35. It should be noted that the best performance in the ArcFace [14] and in the CosFace [41] papers, when used as standalone FR training, are reported using $m = 0.5$ and $m = 0.35$, respectively. It can be also observed from Table 1 that both AdaArcDistill ($m = 0.45$) and AdaCosDistill ($m = 0.35$) achieved very competitive results. This indicates the effectiveness of our AdaDistill on different losses. It should be also noted that AdaDistill does not require tuning any additional hyper-parameters.

Different teacher architectures: The last group of results in Table 1 presents the verification accuracies of MFN models trained with our AdaArcDistill using three different teacher architectures, TransFace-B, ResNet100, ResNet50, and ResNet18. As expected, KD from all teacher models improved the MFN performances. The teachers ResNet100 and TransFace-B, as expected, achieved the highest verification accuracies compared to ResNet50 and ResNet18. However, it can be noticed that ResNet50, as a teacher, leads to better distillation performances, in comparison to the weaker teacher (ResNet18) and stronger teacher (ResNet100 and TransFace-B), which is aligned with findings in [11,30,42]. These results indicate the importance of choosing the right teacher model for KD and the applicability of the AdaDistill concept from teachers with different gaps to the student model.

Identity-disjoint teacher and student training datasets: Given that the class centers in our AdaDistill are calculated based on the exponential moving average of f^t (Eq.6), the student and the teacher, unlike the solution presented in [16, 21, 39], do not necessarily need to share the same training dataset. To validate that, we trained standalone MFN models on Casia-WebFace [45] and IDiff-Face [5], respectively using ArcFace loss. Then MFN models are trained

Method	Training Dataset	LFW	CFP-FP	AgeDB-30	CA-LFW	CP-LFW	Avg.	IJBC 1e-4	IJBC 1e-5
ResNet50 (Teacher)	MS1MV2 [14, 19]	99.80	97.63	97.92	96.10	92.43	96.77	96.05	93.96
MFN (Student)	IDiff-Face [5]	97.05	79.16	81.88	89.37	78.10	85.11	22.19	3.43
MFN + AdaArcDistill		99.22	89.20	92.47	93.85	85.62	**92.07**	**71.33**	**24.83**
MFN (Student)	Casia-WebFace [45]	98.97	93.21	91.40	91.27	86.42	92.25	77.46	57.75
MFN + AdaArcDistill		99.38	94.53	95.23	94.57	90.35	**94.81**	**91.49**	**86.25**

Table 2: Evaluation results of MFN trained with two different datasets, IDiff-Face and CasiaWebFace. The teacher is trained on MS1MV2.

again as a student using our AdaArcFace (Equations 4 6 and 8 and $m1 = 0.45$), and the ResNet50 as the teacher (trained on MS1MV2). Table 2 presents the verification accuracies of the standalone MFN and MFN with AdaArcDistill models. As expected, our AdaArcDistill boosted the MFN verification accuracies under the considered evaluation benchmarks, even though the distillation used different data than the teacher's training data.

5.2 Comparison with SOTA

We compare our AdaDistill with SOTA KD methods on 10 mainstream benchmarks described in Section 4.1. The results of our AdaDistill are reported with two loss functions, noted as AdaArcDisill and AdaCosDistill. Our models are trained only on MS1MV2 [14, 19] for fair comparisons with previous works. Note that not all reported previous KD methods are designed for face recognition [10, 21, 32, 33, 40] or evaluated on the recent face benchmarks in their respective works [1,17,35,36], and thus, we reported the results from [16,24,27,39] when it was available. The evaluation results of our AdaDistill and SOTA KD methods are reported in Table 3 using evaluation metrics described in Section 4.2. Several observations can be made from the reported results in Table 3.

- On the small verification benchmarks, LFW, CFP-FP, AgeDB-30, CA-LFW, and CP-LFW, our AdaArcDstill achived compatitive results to all SOTA techniques, exceeding them in many cases, with the best average accuracy of 95.43% achieved by our AdaArcDstill.
- On the large-scale IJB-C and IJB-B datasets, our AdaArcDstill and AdaCosDstill outperformed all SOTA methods. The best verification accuracies on IJB-C and IJB-B are 93.27% and 91.21% (TAR@FAR1e-4), respectively.
- Our AdaArcDstill and AdaCosDistill demonstrate superiority over the SOTA methods on the recent challenging ICCV21-MFR benchmarks, including all three testing sets, MR-all, Children, and Mask.
- Our AdaArcDstill and AdaCosDistill are ranked among the top-performing methods on the MegaFace and MegaFace (R) benchmarks.
- Compared to the KD method aimed at transferring knowledge from angular space, MarginKD [39] and ShrinkTeaNet [16], our AdaArcDistill and AdaCosDistill outperformed these methods [16, 39] on all benchmarks.
- In comparison to the recent SOTA KD method (ReFO and ReFO+ CVPR2023 [27]) designed for FR, our AdaDistill clearly outperformed ReFO [27] on all large-scale challenging benchmarks, including IJB-C and ICCV21-MFR. As we discussed earlier, our AdaDistill achieved competitive results on MegFace, in comparison to ReFO [27] and other SOTA methods. On small evaluation

Method	LFW	CFP-FP	AgeDB-30	CA-LFW	CP-LFW	Avg.	IJB-C		IJB-B		ICCV21-MFR			MegaFace			
							1e-4	1e-5	1e-4	1e-5	MF-all	Children	Mask	Id(R)	Ver(R)	Id	Ver
IR50 (teacher)	99.80	97.63	97.92	96.05	92.50	96.78	95.16	92.66	93.45	88.65	75.48	49.41	54.50	98.14	98.34	80.62	96.83
MFN (student)	99.52	91.66	95.82	95.12	87.93	94.01	89.13	81.65	87.07	74.63	53.43	24.71	27.90	90.91	92.71	75.52	90.80
FitNet (arxiv14) [36]	99.47	91.30	96.18	95.12	88.30	94.07	87.76	73.71	86.35	70.19	54.46	26.62	28.47	91.16	92.34	75.88	90.64
KD (NIPSW14) [21]	99.50	91.71	95.93	95.03	87.85	94.00	88.37	80.39	86.08	74.30	50.77	26.36	25.74	90.40	92.00	75.81	90.07
DarkRank (AAAI18) [10]	99.55	91.84	95.60	95.07	87.77	93.97	89.28	81.62	86.76	73.75	56.82	28.84	30.07	90.76	92.41	75.80	90.66
SP (ICCV19) [40]	99.53	92.33	96.17	95.07	88.45	94.31	88.43	78.13	86.34	72.85	54.44	26.63	29.75	91.25	92.41	75.37	90.62
CCKD (ICCV19) [35]	99.47	91.90	95.83	95.22	88.48	94.18	87.99	78.75	85.63	72.38	55.64	27.65	30.22	91.17	92.76	75.73	90.63
RKD (CVPR19) [33]	99.58	92.13	96.18	95.25	87.97	94.22	89.65	83.21	87.27	75.17	53.92	27.91	27.94	91.44	92.92	75.73	91.21
ShrinkTeaNet (arxiv19) [16]	99.47	91.97	96.00	94.98	88.52	94.19	87.80	79.78	85.31	75.23	55.28	27.73	30.24	90.73	92.32	75.55	90.56
TripletDistillation (ICIP20) [17]	99.55	93.14	95.53	94.97	88.03	94.24	84.57	76.65	81.88	70.51	-	-	-	86.52	88.75	71.93	91.35
MarginKD (arxiv2020) [39]	*99.61*	92.01	96.55	95.13	88.03	94.27	85.71	75.00	82.97	66.25	50.73	25.14	28.54	91.70	92.96	76.34	91.31
SFTN (NIPS21) [32]	99.48	92.77	96.30	-	-	-	90.96	82.67	-	-	55.50	28.51	29.66	91.69	93.38	-	-
EKD (CVPR22) [24]	99.60	94.33	96.48	*95.37*	*89.35*	95.03	90.48	84.00	88.35	76.60	56.60	28.95	32.14	91.02	93.08	75.54	91.42
SH-KD (arxiv22) [1]	99.47	94.67	96.53	-	-	-	91.75	85.76	-	-	57.69	30.15	32.01	**92.51**	**93.93**	-	-
ReFO (CVPR23) [27]	99.55	94.51	**96.92**	-	-	-	92.23	87.55	-	-	56.63	33.36	31.88	92.38	93.80	-	-
ReFO+ (CVPR23) [27]	**99.65**	94.77	96.42	-	-	-	92.41	87.80	-	-	*59.17*	32.80	32.24	92.41	93.75	-	-
AdaArcDistill (ours)	99.58	**95.24**	*96.78*	**95.55**	**90.02**	**95.43**	**93.27**	**89.32**	**91.21**	*84.13*	58.37	*37.21*	*35.36*	92.12	93.32	**76.39**	**91.54**
AdaCosDistill (ours)	99.50	*95.07*	96.60	95.33	**90.02**	*95.30*	*93.05*	*89.21*	*90.93*	**84.32**	**60.56**	**39.94**	**37.06**	91.05	92.68	75.43	90.83

Table 3: Evaluation results of our AdaDsitll (AdaArcDistill and AdaCosDistill) and various SOTA methods. The results are reported in (%) on the five small benchmarks along with the average accuracy on these benchmarks. Our AdaDsitlls are among the top-performing methods on the small benchmarks with the best overall average accuracy. We also reported the results on the large-scale IJB-C and IJB-B benchmarks where our AdaDsitlls outperformed all SOTA methods. On the recent challenging ICCV21-MFR, our AdaDistill outperformed all SOTA methods and achieved very competitive results on MegaFace. The best result on each dataset is in **bold** and the second is in *italic*. Results are reported using the evaluation metrics described in Section 4.2.

benchmarks, our AdaDistill and ReFO achieved very competitive results. However, unlike ReFO [27], our AdaDistill does not require student proxy or reverse distillation through multi-phases of student-teacher training.

6 Limitation and social impact

Although our AdaDistill boosted the SOTA performances of compact FR, the verification accuracies are still comparably lower than the ones achieved by the computationally expensive SOTA FRs [14,23], which might limit the deployability of such compact models in some scenarios, especially in critical use-cases that require the highest possible verification accuracies. The training of FR might involve collecting, sharing, or distributing authentic data that could be collected without proper user consent. Aiming at mitigating this issue, several recent works [5, 6, 26] proposed the use of synthetic data to train standalone FR models. This work takes a step in this direction and proposes to train and evaluate a student within the KD framework using synthetically generated data.

7 Conclusion

We proposed in this paper a novel Adaptive feature-based KD approach, AdaDistill, for deep face recognition. The proposed AdaDistill targets enhancing the verification accuracies of compact students while taking into consideration the relatively low capacity of shallow student architecture and the importance of the hard samples in the distillation process. This is achieved by building our distillation concept with the margin-penalty softmax while considering the class centers of the teacher. This is achieved by adaptively distilling the class centers of the teacher to the student. The adaptive nature is controlled by the progressing ability of the student to learn complex information over training iterations. Through extensive experiments on mainstream face benchmarks, we demonstrated the effectiveness of our AdaDsitll and its superiority over the SOTA competitors, along with detailed experiments empirically proving our design choices.

Acknowledgments

This research work has been funded by the German Federal Ministry of Education and Research and the Hessen State Ministry for Higher Education, Research and the Arts within their joint support of the National Research Center for Applied Cybersecurity ATHENE and the ARIS Research Programme P2-0250 Metrology and Biometric Systems.

References

1. Baruch, E.B., Karklinsky, M., Biton, Y., Ben-Cohen, A., Lawen, H., Zamir, N.: It's all in the head: Representation knowledge distillation through classifier sharing. CoRR **abs/2201.06945** (2022), https://arxiv.org/abs/2201.06945
2. Boutros, F., Damer, N., Fang, M., Kirchbuchner, F., Kuijper, A.: Mixfacenets: Extremely efficient face recognition networks. In: International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021. pp. 1–8. IEEE (2021). https://doi.org/10.1109/IJCB52358.2021.9484374, https://doi.org/10.1109/IJCB52358.2021.9484374
3. Boutros, F., Damer, N., Kirchbuchner, F., Kuijper, A.: Elasticface: Elastic margin loss for deep face recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2022, New Orleans, LA, USA, June 19-20, 2022. pp. 1577–1586. IEEE (2022). https://doi.org/10.1109/CVPRW56347.2022.00164, https://doi.org/10.1109/CVPRW56347.2022.00164
4. Boutros, F., Damer, N., Kuijper, A.: Quantface: Towards lightweight face recognition by synthetic data low-bit quantization. In: 26th International Conference on Pattern Recognition, ICPR 2022, Montreal, QC, Canada, August 21-25, 2022. pp. 855–862. IEEE (2022). https://doi.org/10.1109/ICPR56361.2022.9955645, https://doi.org/10.1109/ICPR56361.2022.9955645
5. Boutros, F., Grebe, J.H., Kuijper, A., Damer, N.: Idiff-face: Synthetic-based face recognition through fizzy identity-conditioned diffusion model. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 19650–19661 (October 2023)
6. Boutros, F., Struc, V., Fiérrez, J., Damer, N.: Synthetic data for face recognition: Current state and future prospects. Image Vis. Comput. **135**, 104688 (2023). https://doi.org/10.1016/J.IMAVIS.2023.104688, https://doi.org/10.1016/j.imavis.2023.104688
7. Bucila, C., Caruana, R., Niculescu-Mizil, A.: Model compression. In: Eliassi-Rad, T., Ungar, L.H., Craven, M., Gunopulos, D. (eds.) Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, USA, August 20-23, 2006. pp. 535–541. ACM (2006). https://doi.org/10.1145/1150402.1150464, https://doi.org/10.1145/1150402.1150464
8. Caldeira, E., Neto, P.C., Huber, M., Damer, N., Sequeira, A.F.: Model compression techniques in biometrics applications: A survey. CoRR **abs/2401.10139** (2024). https://doi.org/10.48550/ARXIV.2401.10139, https://doi.org/10.48550/arXiv.2401.10139
9. Chen, S., Liu, Y., Gao, X., Han, Z.: Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices. In: Zhou, J., Wang, Y., Sun, Z., Jia, Z.,

Feng, J., Shan, S., Ubul, K., Guo, Z. (eds.) Biometric Recognition - 13th Chinese Conference, CCBR 2018, Urumqi, China, August 11-12, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10996, pp. 428–438. Springer (2018). https://doi.org/10.1007/978-3-319-97909-0_46, https://doi.org/10.1007/978-3-319-97909-0_46

10. Chen, Y., Wang, N., Zhang, Z.: Darkrank: Accelerating deep metric learning via cross sample similarities transfer. In: McIlraith, S.A., Weinberger, K.Q. (eds.) Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018. pp. 2852–2859. AAAI Press (2018), https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17147
11. Cho, J.H., Hariharan, B.: On the efficacy of knowledge distillation. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019. pp. 4793–4801. IEEE (2019). https://doi.org/10.1109/ICCV.2019.00489, https://doi.org/10.1109/ICCV.2019.00489
12. Dan, J., Liu, Y., Xie, H., Deng, J., Xie, H., Xie, X., Sun, B.: Transface: Calibrating transformer training for face recognition from a data-centric perspective. In: IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023. pp. 20585–20596. IEEE (2023). https://doi.org/10.1109/ICCV51070.2023.01887, https://doi.org/10.1109/ICCV51070.2023.01887
13. Deng, J., Guo, J., An, X., Zhu, Z., Zafeiriou, S.: Masked face recognition challenge: The insightface track report. In: IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, BC, Canada, October 11-17, 2021. pp. 1437–1444. IEEE (2021). https://doi.org/10.1109/ICCVW54120.2021.00165, https://doi.org/10.1109/ICCVW54120.2021.00165
14. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. pp. 4690–4699. Computer Vision Foundation / IEEE (2019). https://doi.org/10.1109/CVPR.2019.00482
15. Deng, J., Guo, J., Zhang, D., Deng, Y., Lu, X., Shi, S.: Lightweight face recognition challenge. In: 2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019. pp. 2638–2646. IEEE (2019). https://doi.org/10.1109/ICCVW.2019.00322, https://doi.org/10.1109/ICCVW.2019.00322
16. Duong, C.N., Luu, K., Quach, K.G., Le, N.: Shrinkteanet: Million-scale lightweight face recognition via shrinking teacher-student networks. CoRR **abs/1905.10620** (2019), http://arxiv.org/abs/1905.10620
17. Feng, Y., Wang, H., Hu, H.R., Yu, L., Wang, W., Wang, S.: Triplet distillation for deep face recognition. In: IEEE International Conference on Image Processing, ICIP 2020, Abu Dhabi, United Arab Emirates, October 25-28, 2020. pp. 808–812. IEEE (2020). https://doi.org/10.1109/ICIP40778.2020.9190651, https://doi.org/10.1109/ICIP40778.2020.9190651
18. Gou, J., Yu, B., Maybank, S.J., Tao, D.: Knowledge distillation: A survey. Int. J. Comput. Vis. **129**(6), 1789–1819 (2021). https://doi.org/10.1007/S11263-021-01453-Z, https://doi.org/10.1007/s11263-021-01453-z

19. Guo, Y., Zhang, L., Hu, Y., He, X., Gao, J.: Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III. Lecture Notes in Computer Science, vol. 9907, pp. 87–102. Springer (2016). https://doi.org/10.1007/978-3-319-46487-9_6, https://doi.org/10.1007/978-3-319-46487-9_6
20. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. pp. 770–778. IEEE Computer Society (2016). https://doi.org/10.1109/CVPR.2016.90, https://doi.org/10.1109/CVPR.2016.90
21. Hinton, G.E., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. CoRR **abs/1503.02531** (2015), http://arxiv.org/abs/1503.02531, nIPS 2014 Deep Learning Workshop
22. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Tech. Rep. 07-49, University of Massachusetts, Amherst (October 2007)
23. Huang, Y., Wang, Y., Tai, Y., Liu, X., Shen, P., Li, S., Li, J., Huang, F.: Curricularface: Adaptive curriculum learning loss for deep face recognition. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020. pp. 5900–5909. Computer Vision Foundation / IEEE (2020). https://doi.org/10.1109/CVPR42600.2020.00594, https://openaccess.thecvf.com/content_CVPR_2020/html/Huang_CurricularFace_Adaptive_Curriculum_Learning_Loss_for_Deep_Face_Recognition_CVPR_2020_paper.html
24. Huang, Y., Wu, J., Xu, X., Ding, S.: Evaluation-oriented knowledge distillation for deep face recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022. pp. 18719–18728. IEEE (2022). https://doi.org/10.1109/CVPR52688.2022.01818, https://doi.org/10.1109/CVPR52688.2022.01818
25. Kemelmacher-Shlizerman, I., Seitz, S.M., Miller, D., Brossard, E.: The megaface benchmark: 1 million faces for recognition at scale. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. pp. 4873–4882. IEEE Computer Society (2016). https://doi.org/10.1109/CVPR.2016.527, https://doi.org/10.1109/CVPR.2016.527
26. Kim, M., Liu, F., Jain, A.K., Liu, X.: Dcface: Synthetic face generation with dual condition diffusion model. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023. pp. 12715–12725. IEEE (2023). https://doi.org/10.1109/CVPR52729.2023.01223, https://doi.org/10.1109/CVPR52729.2023.01223
27. Li, J., Guo, Z., Li, H., Han, S., Baek, J.w., Yang, M., Yang, R., Suh, S.: Rethinking feature-based knowledge distillation for face recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 20156–20165 (June 2023)
28. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., Song, L.: Sphereface: Deep hypersphere embedding for face recognition. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 6738–6746. IEEE Computer Society (2017). https://doi.org/10.1109/CVPR.2017.713, https://doi.org/10.1109/CVPR.2017.713

29. Maze, B., Adams, J.C., Duncan, J.A., Kalka, N.D., Miller, T., Otto, C., Jain, A.K., Niggel, W.T., Anderson, J., Cheney, J., Grother, P.: IARPA janus benchmark - C: face dataset and protocol. In: 2018 International Conference on Biometrics, ICB 2018, Gold Coast, Australia, February 20-23, 2018. pp. 158–165. IEEE (2018). https://doi.org/10.1109/ICB2018.2018.00033, https://doi.org/10.1109/ICB2018.2018.00033
30. Mirzadeh, S., Farajtabar, M., Li, A., Levine, N., Matsukawa, A., Ghasemzadeh, H.: Improved knowledge distillation via teacher assistant. In: The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020. pp. 5191–5198. AAAI Press (2020). https://doi.org/10.1609/AAAI.V34I04.5963, https://doi.org/10.1609/aaai.v34i04.5963
31. Moschoglou, S., Papaioannou, A., Sagonas, C., Deng, J., Kotsia, I., Zafeiriou, S.: Agedb: The first manually collected, in-the-wild age database. In: 2017 IEEE CVPRW, CVPR Workshops 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 1997–2005. IEEE Computer Society (2017). https://doi.org/10.1109/CVPRW.2017.250, https://doi.org/10.1109/CVPRW.2017.250
32. Park, D.Y., Cha, M., Jeong, C., Kim, D., Han, B.: Learning student-friendly teacher networks for knowledge distillation. In: Ranzato, M., Beygelzimer, A., Dauphin, Y.N., Liang, P., Vaughan, J.W. (eds.) Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual. pp. 13292–13303 (2021), https://proceedings.neurips.cc/paper/2021/hash/6e7d2da6d3953058db75714ac400b584-Abstract.html
33. Park, W., Kim, D., Lu, Y., Cho, M.: Relational knowledge distillation. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. pp. 3967–3976. Computer Vision Foundation / IEEE (2019). https://doi.org/10.1109/CVPR.2019.00409, http://openaccess.thecvf.com/content_CVPR_2019/html/Park_Relational_Knowledge_Distillation_CVPR_2019_paper.html
34. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019), http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf
35. Peng, B., Jin, X., Li, D., Zhou, S., Wu, Y., Liu, J., Zhang, Z., Liu, Y.: Correlation congruence for knowledge distillation. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019. pp. 5006–5015. IEEE (2019). https://doi.org/10.1109/ICCV.2019.00511, https://doi.org/10.1109/ICCV.2019.00511
36. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015), http://arxiv.org/abs/1412.6550

37. Ruder, S.: An overview of gradient descent optimization algorithms. CoRR **abs/1609.04747** (2016), http://arxiv.org/abs/1609.04747
38. Sengupta, S., Chen, J., Castillo, C.D., Patel, V.M., Chellappa, R., Jacobs, D.W.: Frontal to profile face verification in the wild. In: 2016 IEEE Winter Conference on Applications of Computer Vision, WACV 2016, Lake Placid, NY, USA, March 7-10, 2016. pp. 1–9. IEEE Computer Society (2016). https://doi.org/10.1109/WACV.2016.7477558, https://doi.org/10.1109/WACV.2016.7477558
39. Svitov, D., Alyamkin, S.: Margindistillation: distillation for margin-based softmax. CoRR **abs/2003.02586** (2020), https://arxiv.org/abs/2003.02586
40. Tung, F., Mori, G.: Similarity-preserving knowledge distillation. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019. pp. 1365–1374. IEEE (2019). https://doi.org/10.1109/ICCV.2019.00145, https://doi.org/10.1109/ICCV.2019.00145
41. Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., Liu, W.: Cosface: Large margin cosine loss for deep face recognition. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. pp. 5265–5274. IEEE Computer Society (2018). https://doi.org/10.1109/CVPR.2018.00552
42. Wang, L., Yoon, K.: Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. IEEE Trans. Pattern Anal. Mach. Intell. **44**(6), 3048–3068 (2022). https://doi.org/10.1109/TPAMI.2021.3055564, https://doi.org/10.1109/TPAMI.2021.3055564
43. Whitelam, C., Taborsky, E., Blanton, A., Maze, B., Adams, J.C., Miller, T., Kalka, N.D., Jain, A.K., Duncan, J.A., Allen, K., Cheney, J., Grother, P.: IARPA janus benchmark-b face dataset. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 592–600. IEEE Computer Society (2017). https://doi.org/10.1109/CVPRW.2017.87
44. Yan, M., Zhao, M., Xu, Z., Zhang, Q., Wang, G., Su, Z.: Vargfacenet: An efficient variable group convolutional neural network for lightweight face recognition. In: 2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019. pp. 2647–2654. IEEE (2019). https://doi.org/10.1109/ICCVW.2019.00323, https://doi.org/10.1109/ICCVW.2019.00323
45. Yi, D., Lei, Z., Liao, S., Li, S.Z.: Learning face representation from scratch. CoRR **abs/1411.7923** (2014), http://arxiv.org/abs/1411.7923
46. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters **23**(10), 1499–1503 (2016)
47. Zheng, T., Deng, W.: Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments. Tech. Rep. 18-01, Beijing University of Posts and Telecommunications (February 2018)
48. Zheng, T., Deng, W., Hu, J.: Cross-age LFW: A database for studying cross-age face recognition in unconstrained environments. CoRR **abs/1708.08197** (2017), http://arxiv.org/abs/1708.08197