International Journal of Computer Vision
https://doi.org/10.1007/s11263-024-02080-0

Adaptive Discriminative Regularization for Visual Classification

Qingsong Zhao[1,2] · Yi Wang[3] · Shuguang Dou[1] · Chen Gong[4,5] · Yin Wang[1] · Cairong Zhao[1,2]

Received: 11 January 2023 / Accepted: 13 April 2024

Abstract
How to improve discriminative feature learning is central in classification. Existing works address this problem by explicitly increasing inter-class separability and intra-class compactness by constructing positive and negative pairs for contrastive learning or posing tighter class separating margins. These methods do not exploit the similarity between different classes as they adhere to independent identical distributions assumption in data. In this paper, we embrace the real-world data distribution setting in that some classes share semantic overlaps due to their similar appearances or concepts. Regarding this hypothesis, we propose a novel regularization to improve discriminative learning. We first calibrate the estimated highest likelihood of one sample based on its semantically neighboring classes, then encourage the overall likelihood predictions to be deterministic by imposing an adaptive exponential penalty. As the gradient of the proposed method is roughly proportional to the uncertainty of the predicted likelihoods, we name it adaptive discriminative regularization (ADR), trained along with a standard cross entropy loss in classification. Extensive experiments demonstrate that it can yield consistent and non-trivial performance improvements in a variety of visual classification tasks (over 10 benchmarks). Furthermore, we find it is robust to long-tailed and noisy label data distribution. Its flexible design enables its compatibility with mainstream classification architectures and losses.

Keywords Image classification · Regularization · Loss function · Cross entropy

1 Introduction

Visual classification is one of the fundamental research topics in machine learning and computer vision. Typically, it transforms high-dimensional visual signals (e.g., images, videos, etc.) into the corresponding latent features, and then differentiates them into different classes. With the advance of deep learning and the availability of big data, visual classification makes significant leaps in both theories and practices, widely employed in face recognition, object detection, and so on.

In visual classification studies, efforts have been made to improve discrimination ability by increasing inter-class separability and intra-class compactness. Existing methods (Hadsell et al., 2006; Liu et al., 2016; Zhu et al., 2019; Sun et al., 2020) are either based on positive and negative sample

Communicated by Liwei Wang.

Qingsong Zhao, Yi Wang have equally contributed to this work.

A preprint version of this research work was put on arXiv (Zhao et al., 2022).

✉ Cairong Zhao
zhaocairong@tongji.edu.cn

Qingsong Zhao
qingsongzhao@tongji.edu.cn

Yi Wang
wangyi@pjlab.org.cn

Shuguang Dou
2010504@tongji.edu.cn

Chen Gong
chen.gong@njust.edu.cn

Yin Wang
yinw@tongji.edu.cn

[1] Department of Computer Science and Technology, Tongji University, Shanghai 201804, China

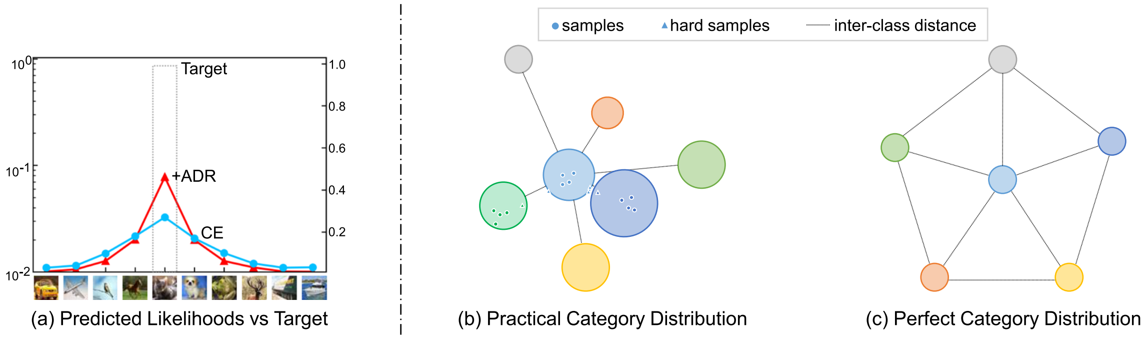
[2] State Key Laboratory of Integrated Services Networks, Xidian University, Xi’an, China

[3] Shanghai AI Laboratory, Shanghai 200232, China

[4] Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information, Ministry of Education, Nanjing 210094, China

[5] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China

Published online: 13 May 2024



(a) Predicted Likelihoods vs Target

(b) Practical Category Distribution

(c) Perfect Category Distribution

Fig. 1 Different predicted logit distributions are plotted in (**a**). Different data distributions are drawn in (**b**) and (**c**). When collecting data in the wild, one often expects the distance between clusters to be equal i.e., (**c**) an ideal category distribution. But the most cases, we collected the dataset can be represented as (**b**), in which the clusters vary in variance and inter-class distance (various sizes of circles indicate different variances). The hard samples at the edge of the clusters contribute significantly to the decision surface (see Toneva et al. (2019)), while they are more likely to make ambiguous predictions as the blue curve is drawn in (**a**)

pairs or large-margin softmax, the former favors representational learning and the latter is an optimization of the decision surfaces. Specifically, the L-softmax (Liu et al., 2016) and its variants (Liu et al., 2017; Wang et al., 2018; Deng et al., 2019) reinforce the deep neural network learning a bigger margin around the separating hyperplanes by factorizing the cosine similarity into amplitude and angular. This idea is equivalent to making hyperplanes with a larger margin than the original ones driven by a vanilla softmax (aka cross entropy) loss. Both these two types of methods usually work decently on fine-grained (e.g., LFW (Huang et al., 2008)) or small-scale visual classification datasets (e.g., CIFAR-10 (Krizhevsky, 2009)), while bringing trivial benefits to large-scale discrimination tasks. We suppose these lifting optimizations' low-bound methods could accelerate large-scale classification training but hardly improve its performance. Because real-world data are not distributed ideally, hard examples (usually found in large-scale datasets) hinder the effective training of large class margins.

Specifically, training and testing data in the existing visual classification are supposed to be in independent identical distributions (i.i.d.). In this scenario, training a classification model by minimizing the cross entropy between the predicted likelihood and the given ground truth can lead to a discriminative representation, guaranteed by maximum likelihood estimation. This usually does not hold true, as several defined categories share similar concepts (e.g., cats and tigers have similar visual appearance) and some ones have large intra-class dispersion (i.e. one single cluster has different similarity to the others as given in Fig. 1b). We suppose that can be a big challenge for the traditional maximum likelihood estimation training strategy built upon i.i.d, see Arora et al. (2018) and Banburski et al. (2021) for similar statements. For this purpose, we propose a new classification regularization on the estimated likelihoods by exploiting inevitable data dependence.

Due to the pervasive inter-class similarities and intra-class dispersion, as shown in Fig. 1a the predicted likelihoods (i.e. predicted logits from softmax) tend to show a smooth distribution instead of a spiky one, contradicting the initial data assumption that each example has only one label. Thus, we propose an adaptive discriminative regularization to encourage the predicted likelihoods to be deterministic and stabilize such optimization procedure by controlling gradient magnitude according to the certainty of likelihoods. Specifically, as shown in Fig. 2, we firstly calibrate the predicted maximum likelihoods of one sample by its semantically similar classes, then we exert a discriminative constraint on the predicted likelihoods based on a normalized exponential function, that does not only makes the corresponding gradients adaptive to the confidence of predicted likelihoods (i.e. high deterministic likelihoods give a small gradient magnitude, while low deterministic likelihoods give a big one), but also is optimization-friendly.

We validate the effectiveness of our hypothesis and the corresponding regularization by visualizing the features learned by different optimization targets, as illustrated in Fig. 3. To quantify the performance of the trained classification models, we employed the expected calibration error (ECE) metric, as proposed by Guo et al. (2017). Our proposed ADR method consistently achieves a lower ECE value compared to other methods, indicating its superior calibration performance. It is worth noting that the inter-class distance achieved with our ADR method is larger than that achieved with the existing baselines. Intuitively, it can bridge the gap between the estimated likelihood and the practical data distribution (i.e. it may have a better separating hyperplane).

We thoroughly discuss the properties of our proposed ADR theoretically. More importantly, we empirically val-

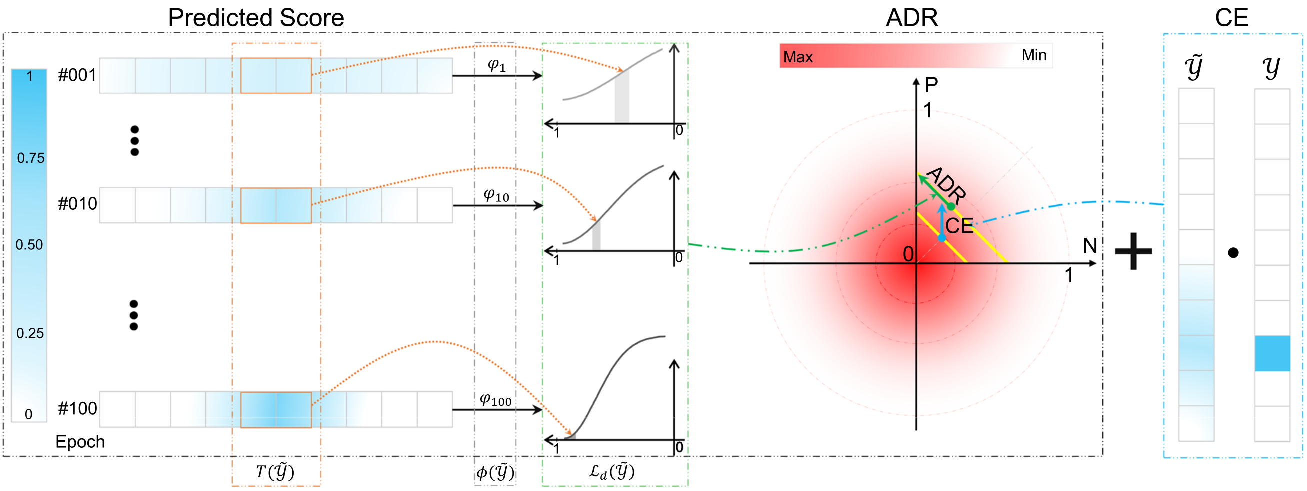


Fig. 2 The conceptual illustration of how training a supervised model with our ADR in a toy experiment (details see Sect. 3.3). Minibatch images are fed into the backbone and softmax to obtain the predicted logits $\tilde{\mathcal{Y}}$. At different training phases, we capture the predicted logits, normalize them according to their confidence, and plot the change curve of ADR (the gray shaded area indicates the possible range of Top-1 logits). The cross-entropy loss pushes the optimization direction (the blue arrow) to the positive side at the beginning, while our ADR enlarges the separability between classes and makes the optimization fall to one side further (the green arrow)

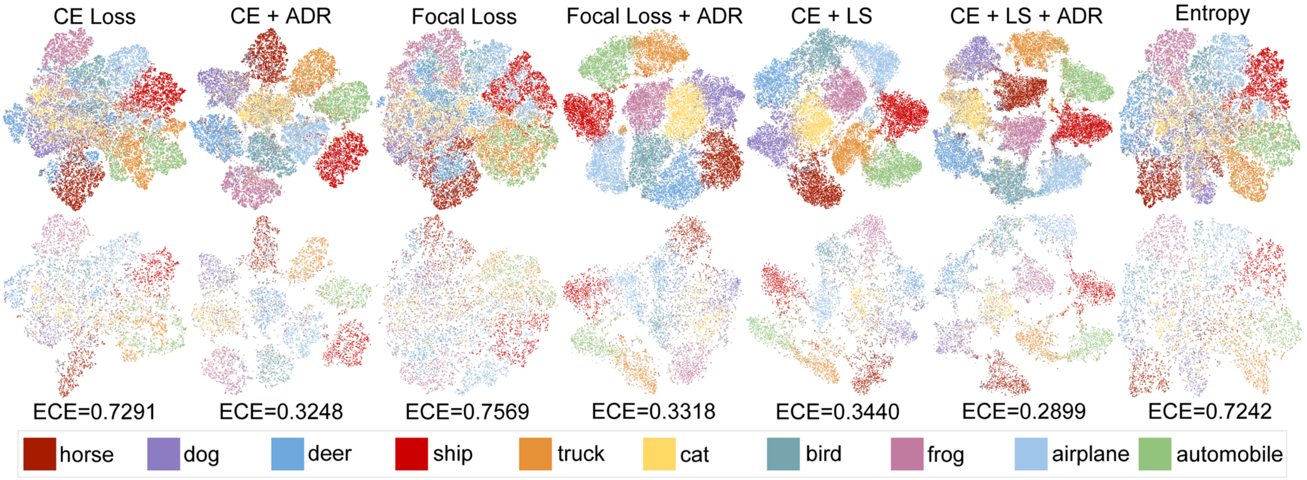


Fig. 3 Visualization of classifier layer's features. The first row comes from a training set, the second row is from a validation set. To be convincing, we plotted the entire sample. The first five columns represent the features extracted from the model which is trained by CE loss, CE w/ ADR, CE w/ LS, CE w/ LS w/ ADR, and CE w/ Entropy loss respectively. The Expected Calibration Error (ECE) on the CIFAR-10 validation set is reported below. Lower is better

idate its effectiveness in multiple mainstream classification benchmarks with various settings in Sect. 4. Extensive experiments show our ADR improves current mainstream model architecture with standard classification optimizations non-trivially. Also, its universal properties make the trained model robust to long-tailed and noise-labeled validation data. In summary, our contributions are summarized as follows:

- We design a new discriminative regularization approach for the supervised visual classification to enlarge the inter-class distance. It is relatively orthogonal to various discriminative optimization targets as it can further improve existing baselines non-trivially.
- We demonstrate that the proposed ADR is compatible with backbones in convolutional neural network (CNN), transformer, and multilayer perceptron (MLP) architectures, and it exhibits robustness against noisy labels and long-tailed distributions.

2 Related Work and Preliminaries

Existing works such as contrastive loss (Hadsell et al., 2006), triplet loss (Schroff et al., 2015), L-Softmax loss (Liu et al., 2016), SphereFace (Liu et al., 2017), Cosface (Wang et al., 2018), Arcface (Deng et al., 2019), online label smoothing (OLS Zhang et al. (2021)) and circle loss (Sun et al., 2020) have been proposed to enhance the performance of traditional softmax cross entropy loss. In this section, we will briefly overview these methods respectively according to their motivations. Specifically, this work is inspired by superset label learning, we will introduce the definition and some research works on it at the end.

Model Regularization

In deep learning, many strategies are known collectively as *regularizations*. In order to reduce the validation error, those strategies often trade off the increase in training error. For example, to alleviate the issue of over-fitting, label smoothing (Szegedy et al., 2016) avoids the search for exact likelihoods. It computes cross entropy with a weighted mixture of uniform distribution of targets (i.e. injecting noise into the targets). Label smoothing (LS) is a technique that chooses to disregard particularly challenging samples in order to effectively capture simpler samples, thereby mitigating the risk of overfitting. As a complement, our ADR focuses on likelihood estimation to improve the confidence of the logits. And, this improvement is contingent upon the assumption that the samples are being optimized in the correct direction. In Müller et al. (2019), Muller et al. discussed why and when label smoothing should work, and demonstrated that label smoothing implicitly calibrates learned models. Arguably, the accuracy improvement is not obvious or even decreases if using relatively small networks and face verification tasks with LS. But in the same case, our ADR still works (please turn to Sect. 4.1 for details).

Discriminatory Feature Learning

For the same motivation (i.e. learning discriminatory features), existing methods are either designed to increase the learning difficulty of separating hyperplanes or require positive and negative samples as training. For example, contrastive loss requires the same class features to be as similar as possible, yet the distance between different class features is larger than a margin. And the triplet loss requires 3 input samples at a time and maximizes the distance between the anchor and a negative sample. Alternatively, the circle loss provides a more flexible optimization approach by assigning distinct penalty strengths for each similarity score. Specifically, it differentiates between the within-class and inter-class similarity scores, allowing for more fine-grained adjustments during optimization. But they all require a carefully designed pair selection procedure. By contrast, the L-Softmax loss was first proposed in a novel view of the cosine similarity to learn discriminative features and bring a series of extension researches (Wang et al., 2018; Liu et al., 2017; Deng et al., 2019). For example, despite the similarity between ArcFace and previous works, it has a better geometric attribute. However, all those methods will increase additional parameters W of fully connected layers compared with the original softmax CE loss. Specifically, to prevent embedding models from learning noisy representations, Shi et al. proposed probabilistic face embeddings (PFEs Shi and Jain (2019)). But it needs additional calculations to estimate a distribution in the latent space. And the Arcface does not converge well with the PFEs. In contrast to those methods, ADR can be embedded into them easily without adding more cost.

Superset Label Learning

The Superset label learning (SLL, aka partial label learning (Wang et al., 2019a; Xu et al., 2021a; Wang et al., 2021)) is a machine learning paradigm that differs from conventional supervised learning, in which one training example can be ambiguously annotated with a set of labels, among which only one is correct. Existing methods for SLL commonly contain an explicit disambiguation operation to pick up the ground truth label of each training example from its candidate labels. For example, Gong et al. (2017) utilizes the l_2^2 norm (similar to the variance-base approach) as the discrimination term, and develops a regularization approach for the instance-based SLL. Yao et al. (2020) designs an entropy-base regularizer as the discrimination term to enhance the discrimination ability of the model. In this paper, we try to formulate an adaptive discriminative regularization for supervised visual classification. Different from SLL, the supervised learning models need a larger backward gradient to enhance the confidence of predicted likelihoods at the beginning, while such a gradient should be small to avoid over-fitting when training is nearly ended.

Preliminaries

In the framework of maximum likelihood estimation (MLE), the cross entropy loss is employed for visual classification (De Boer et al., 2005). Suppose we have n training instances $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ with the ground truth labels $\mathcal{Y} = \{y_1, y_2, \ldots, y_n\}$, and every y_i only has one explicit value $y_i \in \{1, 2, \ldots, c\}$, c is the number of classes. In practice, we often divide the training data into M batches $\mathcal{X} = \{\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_M\}$, and every $\mathcal{X}_m$ $(m = 1, 2, \ldots, M)$ contains B samples. After the feature embedding and the softmax function, every x_i yields one prediction vector $\tilde{y}_i$. The $\tilde{y}_{ij}$ $(j = 1, 2, \ldots, c)$ denotes the predicted probability that the example x_i belongs to the j-th category. The cross-

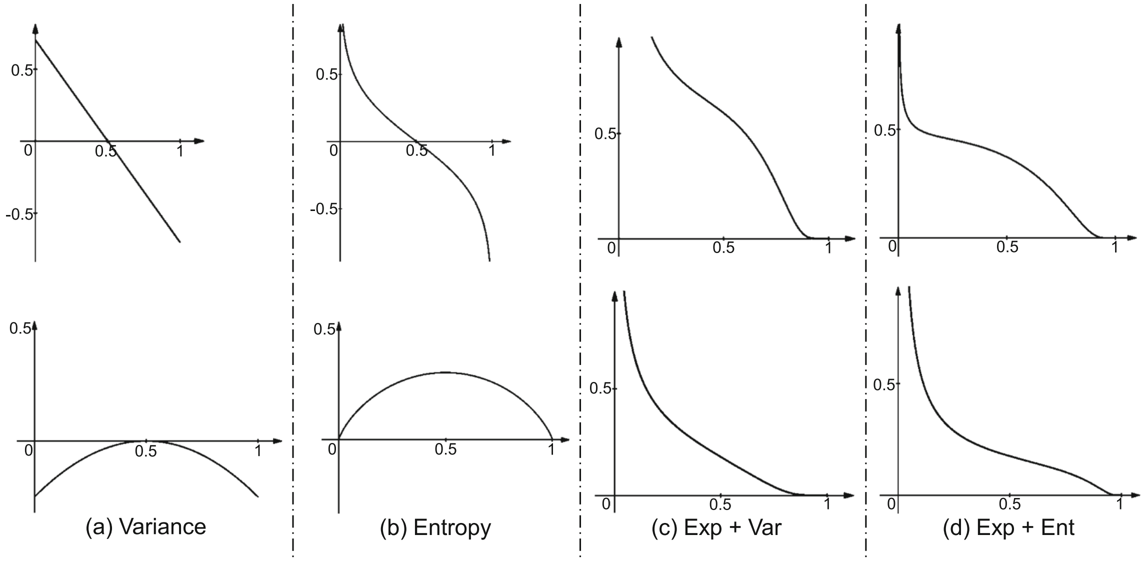


Fig. 4 The visualization of function curves when only the binary classification is considered. The horizontal axis represents the predicted logits (p_t), while the vertical axis represents the value of the corresponding loss function or its derivative. The discriminative function is shown below, and its derivative function is shown above. **a** Variance-base functions. **b** Entropy-base functions. **c** Exponential-variance based functions, a potential solution for ADR. (**d**) As in Eq. (4), exponential-entropy based functions, the solution we choose for ADR

entropy loss for a batch of images can be expressed as

$$\mathcal{L}_{ce}(\tilde{\mathcal{Y}}_m, \mathcal{Y}_m) = -\sum_{i=1}^{B}\sum_{j=1}^{c} y_{ij}\log(\tilde{y}_{ij})) \\ \text{s.t. } \sum_{j=1}^{c} \tilde{y}_{ij} = 1, \tilde{y}_{ij} \geq 0, \forall i = 1, 2, \ldots, B, \tag{1}$$

For simplicity, we also give below the cross entropy and its derivative for the binary case.

$$\begin{cases} \mathcal{L}_{ce}(p_t, y) = -log(p_t) \\ \dfrac{\partial \mathcal{L}_{ce}}{\partial p_t} = -\dfrac{1}{p_t}, \end{cases} \tag{2}$$

where $y \in \{0, 1\}$ specifies the target, p_t donates p if $y = 1$ or $1 - p$ if $y = 0$ ($p \in [0, 1]$). According to Eq. (2), when the p_t approaches 1, the curve of cross entropy flattens out.

3 The Proposed Method

3.1 Intuition

In classical visual classification formulation, each visual sample (image or video) is assigned a discrete label. We then predict the likelihood about this sample belongs to the given classes and infer its estimated label is the one with the maximum logit. Intuitively, the predicted logits should be sparse as only one class is indeed associated. Optimizing the cross entropy between the predicted logits and the ground truth leads the predicted logits to the real data distribution. With this premise, we suppose to encourage the predicted logits to be deterministic will benefit the whole optimization.

Intuitively, we could directly put sparsity regularization on the predicted logits using the approximated implementation, e.g., minimizing l_1 norm of the differential between them, or their entropy. Empirically, we find these solutions can accelerate the complete training but without evident performance increase. We attribute this to their coarse control of the regularization strength. Specifically, the gradient of the l_1 norm is a constant $\mathbb{C}$, while the gradient of the entropy starts with zero and ends with larger values (as shown in Fig. 4b). Neither of them is suitable for the optimization of supervised classification. These observations and analysis inspire us to design an adaptive discriminative regularization. In detail, we expect the penalty on the logits to be large when their distribution seems nearly uniform, while it shrinks to small when their distribution changes to be spiky. Further, considering the optimization efficiency and stability, we prefer the change in the first-order derivative of this penalty to be huge when the predicted logits of each class are close to each other. Meanwhile, such a change gets smaller when these logits differ from each other dramatically.

3.2 Definition

Based on our motivation, we present our adaptive discriminative regularization to improve visual classification optimization. It automatically adjusts the discriminative constraints according to the confidence of the predicted logits. Specifically, we hypothesize the penalties on the predicted likelihoods should be roughly linear to their own uncertainty as plotted in Fig. 4d.

3.2.1 General Form

Our ADR for the supervised classification can be expressed as

$$\begin{aligned} &\mathcal{L}_d(\tilde{\mathcal{Y}}_m) = \sum_{i=1}^{B} \mathcal{F}(\tilde{y}_i), \\ &\text{s.t.} \left|\frac{\partial \mathcal{F}(\tilde{y}_i)}{\partial \tilde{y}_i}\right| := \begin{cases} \approx 0^+, & \phi(\tilde{y}_i) \le \epsilon \\ \propto \phi(\tilde{y}_i), & otherwise \end{cases} \end{aligned} \tag{3}$$

where $\tilde{y}_i \in \tilde{\mathcal{Y}}_m$, and $\varphi_i = \phi(\tilde{y}_i)$ measures the uncertainty of $\tilde{y}_i$ along with the non-negative and normalization constraints ($0 < \varphi_i \le 1$). For example, the entropy-base function satisfying this constraint, able to serve as this uncertainty function ϕ. ϵ is a threshold ($0 < \epsilon < 1$) associated with φ_i and τ (τ is a hyper-parameter in Eq. (4)). $\mathcal{F}(\cdot)$ donates the discriminative regularization function of our ADR. As given in Eq. (3), we expect its partial gradient w.r.t. $\tilde{y}_i$ should be non-negative, and is approximately proportional to the uncertainty of $\tilde{y}_i$ when $\phi(\tilde{y}_i)$ is larger than a given threshold ϵ; while it is smaller than ϵ, such gradient is close to 0.

With this design of gradient changes, it encourages the predicted logits to be certain and the overall training is adaptive based on the certainty of predictions. Specifically, the optimization of visual classification along with this regularization method will be accelerated in the early training stage with the uncertain predicted logits and will be accordingly slowed down when predictions tend to be deterministic. Ideally, this regularization scheme leads to faster convergence and more stable performance during the training plateau.

3.2.2 A Simple Solution

An intuitive simple solution for our ADR (Eq. 3) is given below

$$\begin{aligned} &\mathcal{F}(\tilde{y}_i) = \frac{1}{(\sqrt{2\pi\varphi_i})^{\tau}} \exp\{-\frac{1}{2\varphi_i} T(\tilde{y}_i)\}, \\ &\text{s.t.} \begin{cases} \phi(\tilde{y}_i) = -\frac{1}{B}\sum_{i=1}^{B} \frac{\tilde{y}_i^{\top}\log(\tilde{y}_i)}{\log(c)}, \\ T(\tilde{y}_i) = \|\text{TopK}(\tilde{y}_i, \tau)\|_2^2, \end{cases} \end{aligned} \tag{4}$$

where φ_i is a non-deterministic measure of the predicted vector. We apply the exponential function because of its natural simplicity properties. Different from trigonometric and polynomial functions, all the n-th order partial derivatives of the exponential are their own, and they are always the same increasing functions. Intuitively, the exponential form has better gradient diversity (see Yin et al. (2018)), and that also can simplify the process of backward derivatives (turn to Sect. 3.3 for details). The other basis functions which are subject to Eq. (3) also can be utilized to formulate the solution of our ADR.

$T(\cdot)$ is called *confidence-based normalization* function, characterizing the sufficient statistics (Dynkin, 1978) of the classes based on their predicted logits. TopK$(\cdot)$ is a non-linear sorting function. We suppose that sufficient statistics can be used to alleviate the negative influence brought by bad cases. For example, a classifier may misclassify tigers to cats in some scenarios with certain logits. Due to the apparent similarity between tiger and cat classes, their logits will be close even tiger class has a higher logit. With sufficient statistics, the transformation of the predicted logits will become more uncertain considering local class similarities, giving proper gradient changes even with current wrong predictions. Specifically, a logits vector $\hat{y}_i$ ($\sum_{j=1}^{\tau} \hat{y}_{ij} \le 1$) of the ambiguous classes is selected by TopK$(\tilde{y}_i, \tau)$ from the sparse $\tilde{y}_i$. $\tau \in \mathbb{N}^+$ is a hyper-parameter and indicates the assumed number of similar classes ($1 \le \tau \le c$). The other non-linear functions which can pick out the local class similarities will work too. For example, setting a confidence threshold to generate $\hat{y}_i$. "$\|\cdot\|_2$" computes the l_2 norm of the vector, and the l_1 norm "$\|\cdot\|_1$" may work too. As a result, a simple form of our approach to supervised classification can be written as

$$Loss = \mathcal{L}_{ce}(\tilde{\mathcal{Y}}_m, \mathcal{Y}_m) + \gamma \mathcal{L}_d(\tilde{\mathcal{Y}}_m), \tag{5}$$

where γ is one non-negative trade-off parameter controlling the relative weight of the ADR in the overall cost function.

3.3 Discussion

This section further discusses the necessity of our ADR and clarifies the key operations in it.

Why does minimizing cross entropy need discriminative regularization?

As we described in Sect. 1, classes in real-world datasets are not ideally independent of each other. They share high-level concepts more or less, e.g., cats and dogs both have four legs and fur, compared with airplanes or ships. Simply put, we suppose that supervised tasks are subject to non-deterministic problems too, a discriminative regularization favoring deterministic likelihoods could alleviate this issue. Specifically, we give a conceptual illustration of how cross-entropy and our ADR interact with each other from a toy example as given in Fig. 2. This toy example is conducted on CIFAR-10 (Krizhevsky, 2009) with AlexNet (Krizhevsky, 2014), and our ADR can improve the cross entropy performance up to 4.42% (see Sect. 4.1). At different training phases, we capture the predicted logits and plot their change curve of ADR as given in Fig. 2 (middle). In Fig. 2, we observe that our adaptive discriminative regularization and orthodox cross entropy do **not** share the same optimization path in training. The cross-entropy can be seen as the blue (upward) arrow in Fig. 2, and its optimization direction is pulled to the positive side easily with the MLE algorithm. Given the right optimization direction, we suppose our given ADR can further pull the optimization toward one end along the (yellow) line where the predicted logits are located. In Fig. 2, we find the predicted likelihoods are amplified with ADR, making the optimization quickly fall to one side further. Hence, discriminative regularization is not only suitable for supervised tasks but also requires fine-grained design.

How About Employing Entropy as a Discrimination Term?
We first overview the stochastic gradient descent (SGD) algorithm (Qian, 1999). It follows the estimated gradient downhill as

$$\begin{cases} \mathcal{G} = \dfrac{\partial \mathcal{L}(\theta)}{\partial \theta} \\ \theta \leftarrow \theta - \alpha \mathcal{G}, \end{cases} \tag{6}$$

where θ denotes the learnable parameters, α is the learning rate, $\mathcal{G}$ is the estimated gradient descent of the cost functions. The SGD updates θ by calculating the partial derivatives of the cost function at each parameter, i.e., $\nabla_\theta \mathcal{L}(\theta)$, and it often finds a low value of the cost function quickly. We find that the learning step of every update of θ is **not** proportional to the value the loss function takes, but the partial derivative it makes. From this insight, the existing discrimination terms only pay attention to the functionality of the regularization constraints, i.e., the largest $\mathcal{L}(\theta)$ corresponds to the most ambiguous θ, and vice versa. However, we advocate that in addition to the above criteria, the timing and magnitude of the regularization intervention should be taken into account specifically.

Different discriminative functions are drawn in Fig. 4. Observing the derivative function curves, we find that different from the entropy function, the proposed exponential-based solutions yield a larger gradient for the early to mid-stage of training. Additionally, to prevent over-confident predictions the gradient of our solution rapidly closes to zero later in training. Utilizing the entropy as a discrimination term (e.g., Li et al. (2003); Yao et al. (2020)) to widen the gap of predicted likelihoods could work well in SLL. Because the SLL aims to the problem that a training example is associated with a set of candidate labels. In detail, the gradient of entropy could be zero as we do not know whether the optimization direction is right at the beginning, and it holds a large value due to the model already knowing where the positive side is later in training. Therefore, employing entropy as the discriminative regularization in Eq. (5) will lead to severe optimization issues theoretically and empirically, while the given exponential-based solutions will not.

For the uncertainty function ϕ in Eq.* (4), *entropy-base better than variance-base?
The variance describes the variation of one random variable while the entropy represents the uncertainty of the information, both of them can be used as an uncertainty function ϕ which is designed to evaluate the confidence of predicted likelihoods. However, when a random variable obeys a non-convex distribution, the ability of the variance to describe the information uncertainty will reduce, while the entropy could do better (turn to Zidek and van Eeden (2003) for details). As shown in Fig. 4, we give two specific implementations for it, in which the derivative of the exponential variance decreases faster than that of the exponential entropy. We suppose that the effective interval of its derivative function is then small. Hence, we pick the latter (the exponential entropy in Fig. 4 (d)) as the default solution of our ADR throughout this paper.

Gradient of ADR
Not only does the ADR suit our requirements for its changes, but also it is easy to be optimized according to its gradient form. According to Eq. 4, the solution of our ADR for a single sample $\mathcal{L}_d(\tilde{y}_i)$ can be rewritten as

$$\mathcal{L}_d(\tilde{y}_i) = \prod_{j=1}^{\tau} \frac{1}{\sqrt{2\pi\varphi_i}} \exp\left\{-\frac{1}{2\varphi_i}\hat{y}_{ij}^2\right\}. \tag{7}$$

We only calculate the partial derivative $\frac{\partial \mathcal{L}_d(\tilde{y}_i)}{\partial \tilde{y}_i}$ for simplicity, and it can be computed via

$$\begin{aligned} \frac{\partial \mathcal{L}_d(\tilde{y}_i)}{\partial \tilde{y}_i} &= \sum_{j=1}^{\tau} \left[\mathcal{L}_d(\tilde{y}_i) \cdot \frac{\hat{y}_{ij}^2 \varphi_{ij}' - 2\hat{y}_{ij}\varphi_i - \varphi_i \varphi_{ij}'}{2\varphi_i^2} \right], \\ \text{s.t. } \varphi_{ij}' &= \frac{\partial \varphi_i}{\partial \hat{y}_{ij}}. \end{aligned} \tag{8}$$

The backward derivation (above equation) contains the results of the forward propagation calculation (e.g. Equation (7)), which will reduce the time complexity of our ADR significantly. Specifically, the naive computation of Eq. (8) requires only $O(\tau^2)$ operations, as the terms φ_i and $\mathcal{L}_d(\tilde{y}_i)$ can be computed once and reused in each derivation.

4 Experiments

To evaluate the proposed ADR, we conduct extensive experiments on five typical vision applications, including image

Table 1 Recognition Top-1 error rate on ImageNet-1K classification benchmark

Model	Param	Method	Epochs	Top-1%
RN-50	25.6M	CE loss	250	23.68†
RN-50	25.6M	LS (Szegedy et al., 2016)	250	22.82†
RN-50	25.6M	CutOut (DeVries & Taylor, 2017)	250	22.93†
RN-50	25.6M	BYOT (Zhang et al., 2019)	250	23.04†
RN-50	25.6M	$Tf\text{-}KD_{self}$ (Yuan et al., 2020)	90	23.59†
RN-50	25.6M	$Tf\text{-}KD_{reg}$ (Yuan et al., 2020)	90	23.58†
RN-50	25.6M^{+}	OLS (Zhang et al., 2021)	250	22.28†
RN-101	44.7M	CE loss	250	21.87†
RN-101	44.7M	LS (Szegedy et al., 2016)	250	21.27†
RN-101	44.7M	CutOut (DeVries & Taylor, 2017)	250	20.72†
RN-101	44.7M^{+}	OLS (Zhang et al., 2021)	250	20.85†
I-V2	23.9M	CE loss	–	23.10‡
I-V2	23.9M	LS (Szegedy et al., 2016)	–	22.80‡
I-V4	43.0M	CE loss	–	19.10‡
I-V4	43.0M	LS (Szegedy et al., 2016)	–	19.10‡
RN-50	25.6M	CE loss	100	23.06
RN-50	25.6M	w/ ADR	100	22.49
RN-101	44.7M	CE loss	100	21.30
RN-101	44.7M	w/ ADR	100	20.76
ViT-B/16	86.6M	CE loss	300	18.12
ViT-B/16	86.6M	w/ ADR	300	17.80
ViT-B/16	86.6M	LS	300	18.05
ViT-B/16	86.6M	w/ ADR	300	**17.69**

† and ‡ denote the results reported in Zhang et al. (2021) and Müller et al. (2019) respectively

$^{+}$ means the addition of some parameters

“ResNet” is abbreviated as “RN”, and “I-V2” means “INCEPTION-V2”

classification (ImageNet-1K (Russakovsky et al., 2015), Flowers-102 (Nilsback & Zisserman, 2008), and CIFAR-10), face verification (CASIA (Yi et al., 2014), etc.), facial emotion recognition (FER2013 (Goodfellow et al., 2013)), action recognition (NTU RGB+D Shahroudy et al. (2016)), and unsupervised image segmentation (PASCAL VOC 2012 (Everingham et al., 2015) BSDS500 (Arbelaez et al., 2010)).

Experimental Settings

In all the experiments, we use the same neural network architecture and experimental environment (Pytorch 1.7.0 on NVIDIA 1080Ti) for fair comparisons. Different losses are employed to outline the properties of ADR. The applied neural models include such CNN-based ones with different depths and structures as AlexNet (Krizhevsky, 2014), VGGNet (Khaireddin & Chen, 2021), ResNet-50/101 (He et al., 2016) and the extended ResNet3D-34 (Ji et al., 2021). Transformer-based architectures are evaluated, e.g., ViT (Dosovitskiy et al., 2021) and ConvMixer (Trockman & Kolter, 2022), as well. In the tables, bold formatting signifies the best performance method for the experiment, unless stated otherwise.

4.1 Image Classification

ImageNet-1K

We employ ResNet-50/101 in Radosavovic et al. (2020) as backbones and perform all experiments by utilizing the same training/testing protocols as in Szegedy et al. (2016) and Radosavovic et al. (2020). We also use a vision transformer architecture ViT-B/16 (Dosovitskiy et al., 2021) as the backbone and follow the DeiT (Touvron et al., 2021) training configuration for training. The results are reported in Table 1. Our ADR consistently decreases the error rate on ImageNet-1K with ResNet-50/101 more than 0.5% (0.57% with ResNet-50 and 0.54% with ResNet-101). That validates the effectiveness of the proposed ADR on a large-scale supervised classification dataset with a decently large convolution-based model. Also, such effectiveness is further verified with a popular vision transformer model, ViT-B/16, resulting in a reduction of the error rate by 0.32% and 0.36% compared to the CE and LS methods, respectively. In summary, it shows the performance improvement brought by ADR is relatively agnostic to model architecture.

Table 2 Recognition accuracy on Flowers-102 with the architecture of ResNet-50

Model	Method	Pub.'Year	Top-1%	Top-5%
RN-50	CE loss	TIP'21	90.69†	97.57†
RN-50	CE+LS	TIP'21	92.42†	98.07†
RN-50	CE+OLS	TIP'21	92.86†	98.45†
RN-50	CE loss	TIP'21	90.89	97.45
RN-50	CE+LS	TIP'21	92.12	97.71
RN-50	CE+OLS	TIP'21	93.12	98.31
RN-50	CE+ADR	-	92.36 (1.47 ↑)	98.10
RN-50	LS+ADR	-	93.12 (1.00 ↑)	98.28
RN-50	OLS+ADR	-	**93.49** (0.37 ↑)	98.05

† denotes the results reported in Zhang et al. (2021). ResNet is abbreviated as RN

Flowers-102

We find ADR works fine for fine-grained discriminative tasks. We conducted experiments by following the same architecture of ResNet-50 as Zhang et al. (2021). The results are given in Table 2. They demonstrate that our ADR can handle fine-grained classification. Based on the standard CE, LS, and OLS, the additional ADR can achieve a notable improvement by 1.47%, 1.00%, and 0.37%, respectively.

CIFAR-10

Also, we employ the neutered version of AlexNet (without batch normalization layers Krizhevsky (2014)) and one advanced Transformer-based model ConvMixer as the backbones. We refer to the commonly used protocols with data augmentation in Lee et al. (2015) for training. Quantitative results are shown in Table 3. Compared to the CE, Entropy, and LS, the proposed ADR obtained the best performance boost on both very different backbone baselines (i.e. 4.42% with AlexNet and 2.68% with ConvMixer).

The entropy loss can be embedded in the cross entropy too, we conduct experiments to compare our ADR against it. Following the entropy formulation defined in Yao et al. (2020), we define the total loss function as:

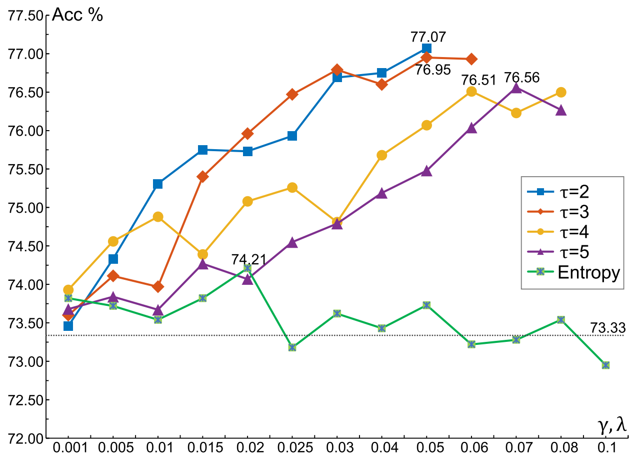


Fig. 5 The classification performance of ADR and the entropy loss, varying γ, λ for ADR (w. optimal τ) and the entropy loss respectively

$$Loss = \mathcal{L}_{ce}(\tilde{\mathcal{Y}}_m, \mathcal{Y}_m) + \lambda \mathcal{L}_e(\tilde{\mathcal{Y}}_m), \tag{9}$$

where λ denote the strength of the modulating term of $\mathcal{L}_e(\tilde{\mathcal{Y}}_m)$. As shown in Table 3, there is a clear gap between the best accuracy of the entropy (74.21%) and ADR (77.07%).

4.1.1 Ablation Study

Specifically, we investigate the influences of hyper-parameters and tolerance of ADR to noisy labels on CIFAR-10.

Sensitivity to Parameters

The tradeoff parameter γ is to control the strength of the proposed ADR. With larger γ, the discriminative force in Fig. 2 becomes larger and the predicted logits become sparser. Another parameter τ closely determines the value of the threshold ϵ in Eq. 3. With smaller τ, the discriminative force will increase and the strength of ADR γ should be smaller accordingly. The relation between the accuracy and γ with different τ on CIFAR-10 is reported in Fig. 5. With the increase of γ and a fixed τ, one can see that the accuracy

Table 3 Recognition accuracy of different methods on CIFAR-10. To demonstrate the effectiveness of ADR even with a tiny model, we utilize the original AlexNet (Krizhevsky, 2014) as the backbone

Model	Method	Pub.'Year	Top-1%	Top-5%
AlexNet†	CE loss	NIPS'19	86.80†	–
AlexNet†	LS	NIPS'19	86.70†	–
AlexNet	CE loss	arXiv'14	73.33	98.01
AlexNet	CE+Entropy	–	74.21 (0.88 ↑)	98.02
AlexNet	CE+ADR	–	77.07 (3.74 ↑)	98.00
AlexNet	LS	CVPR'16	76.31 (2.98 ↑)	98.09
AlexNet	LS+ADR	–	77.75 (4.42 ↑)	97.60
ConvMixer	CE loss	arXiv'22	89.79	99.70
ConvMixer	CE+ADR	–	**92.47** (2.68 ↑)	**99.78**

† denotes the model and results are described in Müller et al. (2019). These results also suggest that LS performs poorly on small datasets and small models

Table 4 The top-1 accuracy comparison of CE and ADR on CIFAR-10 with adding different noise rates (NR)

Method	NR=20%	NR=40%	NR=60%	NR=80%
CE loss	72.51	68.95	63.45	43.38
CE+ADR	**74.89** (2.38 ↑)	**70.53** (1.58 ↑)	**63.65** (0.20 ↑)	**43.83** (0.45 ↑)

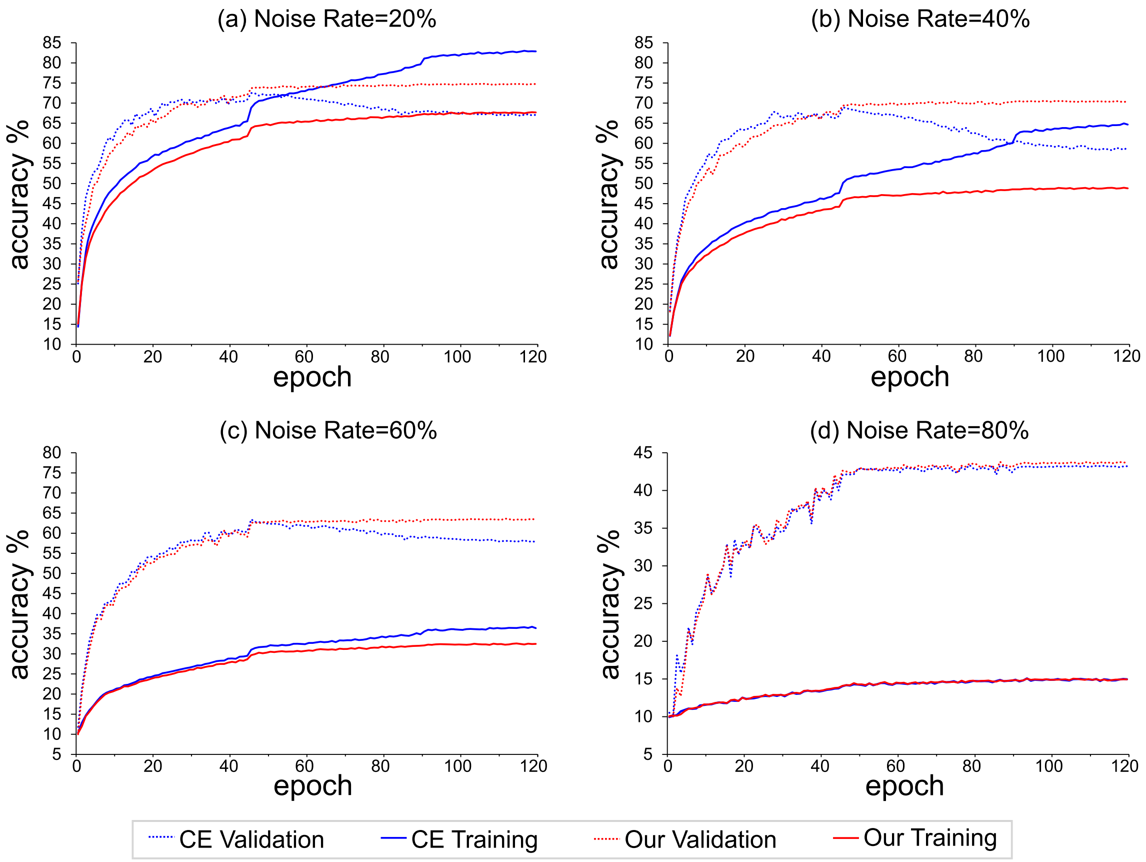


Fig. 6 Classification accuracy under different noise rates. The accuracy curve of ADR stabilizes quickly and does not exceed the percentage of clean labels at convergence, while the CE does not exhibit the same behavior. This observation provides support for the robustness of convergence during ADR training

of ADR is always on the rise. The empirical value for the hyperparameters γ and τ are both related to the distribution of the particular dataset, usually $\tau \ll c$ and $0.01 < \gamma < 0.1$. Still and all, the performance of those parameter settings outperforms the baseline 73.33% (the gray dotted line) by at least 0.13%. Furthermore, the effectiveness of parameter λ (the green one) for the Entropy loss is also drawn in Fig. 5. We see that the accuracy-λ curve of the Entropy is jittering up and down around the baseline. This observation is consistent with the discussion in Sect. 3.3 of the article, where we provide a detailed explanation of why the Entropy function regularity is not suitable for the optimization process in supervised learning.

Tolerance to Noisy Labels

We present how ADR reacts to noisy labels in classification. Generally, training with ADR will prevent the model to overfit noisy distribution. We conducted the experiments using the same settings as Wang et al. (2019b). A certain number of samples are randomly selected and flipped to the uncorrected labels before training. The Top-1 recognition accuracy results under four noisy rates (20%, 40%, 60%, 80%) are reported in Table 4. In addition, the training and test accuracy vs. iteration is visualized in Fig. 6. When an added noise rate is less than 50%, our ADR could obtain a more stable improvement than the CE loss. Note when noise rates exceed 20% or 40%, CE causes training overfitting as the training and test accuracy curves are intersected, and its training accuracy exceeds 80% and 60% respectively. In contrast, the proposed ADR enables the model to be trained robustly.

However, when adding a high noise rate ($NR > 50\%$), the model with CE loss would be under-fitted as more than half training samples are useless or distracting. ADR also fails with heavy noise. After all, in heavy noise conditions, the calibration step (confidence-based normalization) in ADR is

Table 5 Face datasets for training and validation

Datasets	#Identity	#Image
LFW (Huang et al., 2008)	5749	13233
CASIA (Yi et al., 2014)	10K	0.5M
CFP-FP/FF (Sengupta et al., 2016)	500	7000
AgeDB-30 (Moschoglou et al., 2017)	568	16488
CALFW (Zheng et al., 2017)	5749	12174
CPLFW (Zheng & Deng, 2018)	5749	11652

meaningless as there barely exists reliable semantically similar classes. It is worth noting that at 60% noise rates, our ADR still plays a role in counteracting the crossover phenomenon. We think that the 0.2% or 0.45% boost from ADR when noise is greater than 50% may be some sort of random fluctuation in the case of poor CE performance. Therefore, when the proportion of noisy samples increases, it becomes necessary to reduce the hyperparameter γ.

4.2 Face Verification

ADR works fine in face verification. We utilize the same settings in Arcface for the following experiments[1]. The ADR was embedded into the architecture of ResNet18. For model training, we employ the commonly used web-collected outside dataset CASIA (Yi et al., 2014) (excluding the images of identities appearing in the test set) which has $\sim 0.5M$ face images belonging to $\sim 10K$ different individuals.

For the validation, six datasets including LFW, CALFW, CFP-FF, CPLFW, CFP-FP, and AgeDb-30 are utilized to evaluate the performance. LFW includes $\sim 13K$ web-collected images from $\sim 5K$ different identities, with limited variations in pose, age, expression, and illuminations. CPLFW was collected from LFW with a larger pose gap. Similar to CPLFW, CALFW was selected from LFW with higher variations of age. CFP consists of collected images of celebrities in frontal and profile views, which has two evaluation protocols consisting of CFP-Frontal-Frontal and CFP-Frontal-Profile which is a more challenging protocol with around a 90° pose gap within positive pairs. AgeDB-30, a "in-the-wild" dataset, contains manually annotated images. In this paper, we employ the evaluation protocol with a 30-year gap. Table 5 lists the details of these datasets.

4.2.1 Ablation Study

Table 6 presents the results of ADR on such common datasets as LFW, CALFW, CFP-FF, CPLFW, CFP-FP, and AgeDb-30. For LFW and CFP-FP, ADR can boost the accuracy with any γ, τ settings, raising by 0.017%−0.214% compared with both baselines. One can see that ADR can boost the performance over the baselines on CALFW (0.117% and 0.234% respectively). And ADR can further reduce the error rates from $\approx$ 0.6% to $\approx$ 0.5% on CFP-FF. Specifically, ADR can outperform both the baselines by obvious margins (0.683% on AgeDb-30 and 0.700% on the challenging CPLFW respectively).

4.2.2 Comparison with SOTA

We evaluate ADR on serveral state-of-the-art (SoTA) face verification methods, including TigthROI (Xu et al., 2021b), (R+D)BM (Cao et al., 2020) and FAPS$_C$ (Xu et al., 2021b) etc. They are introduced in Table 6. The performance of our ADR is superior to all other losses on LFW, CALFW, CFP-FF/FP, and the average accuracy of those 6 datasets. In conclusion, our ADR can further improve current SOTA face verification results notably.

4.3 Facial Emotion Recognition

The **long-tailed** FER2013 dataset contains $\sim 36K$ images. It has 7 emotion classes, i.e., anger, neutral, disgust, fear, happiness, sadness, and surprise. We employ a customized VGGNet (Khaireddin & Chen, 2021) with an SGD optimizer to conduct experiments, following the official protocols in Goodfellow et al. (2013). In practice, we use the ReduceLROnPlateau[2] against other schedulers to obtain a robust baseline as reported in Table 7. Our ADR is superior to both the baselines of the test and validation sets by 0.6% and 0.57% respectively.

As shown in Fig. 7a, we observe that the model tuned with the vanilla CE overfits the training data, as the training loss continues to decline, while the validation loss is increasing. In our option, to further reduce the training loss, CE focuses on fitting well-classified samples. That is, the score of class label $y = 1$ in the validation set is reduced but still gives the correct prediction results. In contrast, the proposed ADR continues to decline until being stable in both training and validation sets. Note that the training loss of the ADR is larger than CE, but the validation accuracy is greater than it. It empirically validates our ADR focuses on fitting the hard samples (which are misclassified early) and can alleviate the overfitting in training.

4.4 Action Recognition

To verify the effectiveness of the proposed ADR for the sequence-based action recognition (Kong & Fu, 2022), we

[1] The InsightFace project: https://github.com/deepinsight/insightface.git

[2] Pytorch 1.9.0 documentation: https://pytorch.org/docs/stable/_modules/torch/optim/lr_scheduler.html

Table 6 Face Verification performance of different methods on LFW, CALFW, CPLFW, AgeDB-30, CFP-FP and CFP-FF datasets with ResNet18

Method	Pub.'Year	LFW%	CALFW%	CPLFW%	AgeDB-30%	CFP-FF%	CFP-FP%	Average%
CircleLoss (Sun et al., 2020)	CVPR'20	95.167†	84.450†	73.817†	80.867†	96.071†	78.557†	84.822
RBM (Cao et al., 2020)	CVPR'20	99.10‡	91.00‡	87.10‡	91.30‡	-	-	92.125
DBM (Cao et al., 2020)	CVPR'20	99.20‡	92.00‡	87.30‡	91.90‡	-	-	92.600
R&D-BM (Cao et al., 2020)	CVPR'20	**99.30**‡	92.50‡	**87.60**‡	92.10‡	-	-	92.875
Arcface (Deng et al., 2019)	CVPR'19	99.10∗	89.05∗	78.43∗	93.18∗	–	–	89.940
MFR (Guo et al., 2020)	CVPR'20	99.12∗	89.45∗	79.22∗	93.30∗	–	–	90.273
TigthROI (Xu et al., 2021b)	AAAI'21	99.02∗	88.78∗	79.30∗	93.73∗	–	–	90.208
SuperROI (Xu et al., 2021b)	AAAI'21	99.18∗	88.80∗	79.22∗	93.38∗	–	–	90.145
FAPS$_C$ (Xu et al., 2021b)	AAAI'21	99.20∗	89.47∗	80.28∗	**94.02**∗	-	-	90.743
Cosface (Wang et al., 2018)	CVPR'18	99.100	93.033	86.783	93.167	99.429	92.871	93.021
w/ ADR	-	**99.300**	93.267	87.483	93.750	**99.529**	93.057	**93.450**
Arcface (Deng et al., 2019)	CVPR'19	99.200	93.300	86.833	93.267	99.414	93.343	93.150
w/ ADR	-	99.283	**93.417**	87.050	93.950	**99.529**	**93.557**	93.425

‡ and ∗ denote the results reported in Cao et al. (2020) and Xu et al. (2021b), respectively

† denotes the result of our own replication using the Arcface's codebase, following the experimental setup of the circle loss as outlined in Sun et al. (2020)

Table 7 Recognition accuracy of test and validation data on FER2013 dataset

Method	Test %	Validation %
CE loss	72.12	73.82
CE+ADR	**72.72** (0.60 ↑)	**74.39** (0.57 ↑)

Table 8 Recognition accuracy of cross-subject and cross-view evaluations on NTU RGB+D

Method	X-Sub %	X-View %
CE loss	87.87	90.82
CE+ADR	**88.56** (0.69 ↑)	**91.37** (0.55 ↑)

employed the same experiment settings of ResNet3D-34 (Ji et al., 2021) (a Spatio-temporal network). The SGD optimizer with fixed momentum of 0.9 and weight decay of 10^{-5} was utilized for training. We set the LR at 0.01, and it decreases to one-tenth times every 20 epochs.

We apply the NTU RGB+D dataset as a benchmark, which consists of 60 classes and contains 56880 action sequences captured with three cameras from different views. And we follow the cross-subject (X-Sub) and cross-view (X-View) protocols introduced by Shahroudy et al. (2016) to conduct all experiments. The results are reported in Table 8, for the cross-subject, our ADR yields an obvious improvement over the softmax CE loss (0.69%). Additionally, the ADR achieves a noteworthy increase over the softmax CE loss in the protocol of cross-view (0.55%). One can observe that the proposed

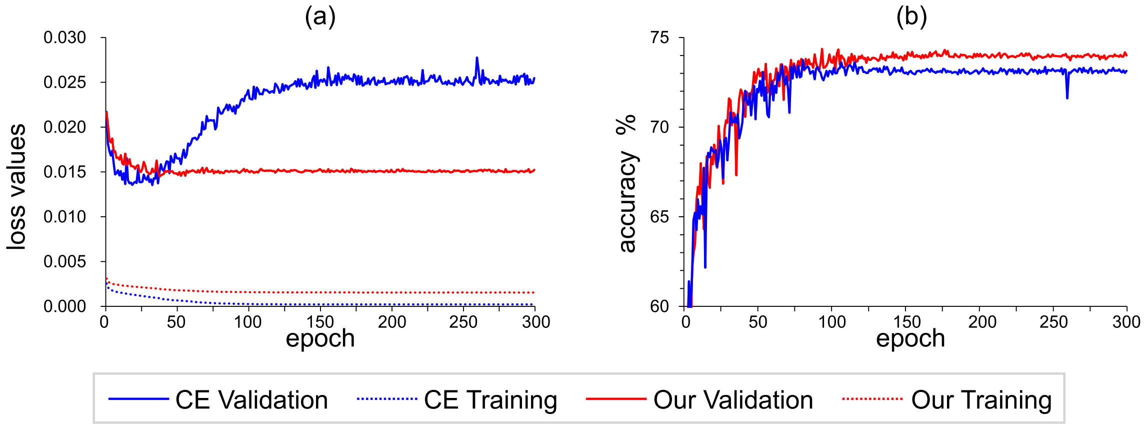


Fig. 7 Recognition loss values and accuracy vs. epoch on FER2013 with different methods. See (**a**), the loss value of ADR decreases rapidly during the first 50 epochs and then gets convergence gradually on the long-tailed dataset

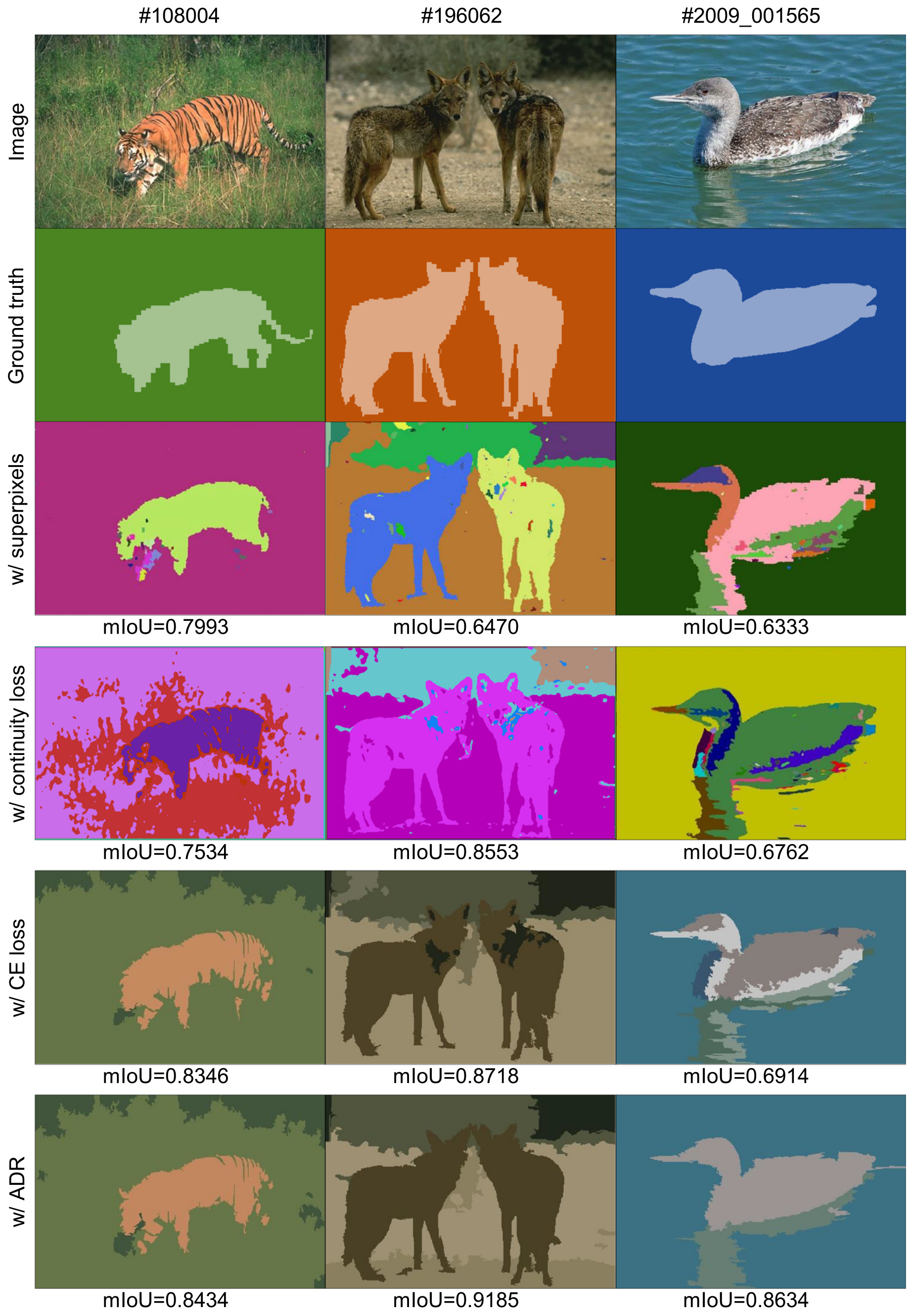


Fig. 8 Qualitative results of the baseline and ours for unsupervised image segmentation on BSDS500 (Arbelaez et al., 2010) and PASCAL VOC 2012 (Everingham et al., 2015). The original images are in the first row, the second consists of the ground truth images, the third contains the results of "superpixels" (Kanezaki, 2018), the fourth shows the results of continuity loss (Kim et al., 2020), the fifth presents the results of CE, and our segmentation results are in the last row. We refer to Kim et al. (2020) to calculate the mIoU metrics and report it below the picture. Different segments are shown in different colors

ADR can yield consistent boosts in the Spatiotemporal architectures.

4.5 Unsupervised Image Segmentation

As mentioned in Sect. 1, our ADR could be applied to several unsupervised learning tasks. Herein, we conduct an exploration namely unsupervised image segmentation by following a pioneer work (Kim et al., 2020). For this task, γ is chosen from $[0.1, 0.5]$ and τ is set as $\{7, 9\}$. Figure 8 presents the qualitative comparison between our ADR and the other methods. Both in the woof (second column) and waterfowl (third column) picture, more precise segments with various colors and textures are detected by our ADR. More specifically, compared to other methods, the model trained with ADR demonstrates the ability to group pixels within woof and waterfowl objects into a single category.

5 Conclusion

In this paper, we propose a practical regularization named adaptive discriminative regularization (ADR) for visual classification, verified on multiple datasets with a spectrum of the model architecture. It is built upon a hypothesis that we can leverage the data with similar semantics to calibrate the predicted likelihood in the target class and encourage it to be assertive, as data (e.g. images or videos) collected in the real world are not ideally independent of each other. Regarding this, ADR considers the intervention timing and gradient magnitude, exactly suitable for supervised visual classification tasks. We give a general form and a simple solution to our method, and those qualities make it compatible with a wide range of visual applications. Moreover, we demonstrate its availability and flexibility by conducting *5* different visual classification tasks on over *10* benchmarks. It also has potential value in objective function designing and other visual applications (e.g., collaborative learning (Du et al., 2022), multi-label learning (Wu et al., 2018), and clustering (Castellano & Vessio, 2022)).

Appendix

Our adaptive discriminative regularization loss for one sample can be written as

$$\begin{aligned}\mathcal{L}_d(\tilde{y}_i) &= \prod_{j=1}^{\tau} \frac{1}{\sqrt{2\pi\varphi_i}} \exp\left\{-\frac{1}{2\varphi_i}\hat{y}_{ij}^2\right\}.\\ &= \prod_{j=1}^{\tau} \mathcal{F}_j(\hat{y}_{ij}),\end{aligned} \tag{10}$$

where φ_i is a function of $\tilde{y}_i$, $\hat{y}_i$ is generated by a non-linear function TopK$(\tilde{y}_i)$. The function $\mathcal{F}_j(\hat{y}_{ij})$ in Eq. (10) can be denoted as

$$\mathcal{F}_j(\hat{y}_{ij}) = \mathcal{H}(\hat{y}_{ij})\mathcal{E}(\hat{y}_{ij}), \tag{11}$$

where $\mathcal{H}(\hat{y}_{ij})$ is called the base measure function "$\frac{1}{\sqrt{2\pi\varphi_i}}$", $\mathcal{E}(\hat{y}_{ij})$ is named the exponential term "$\exp\{-\frac{1}{2\varphi_i}\hat{y}_{ij}^2\}$".

In the backward propagation, $\frac{\partial\mathcal{L}_d(\tilde{y}_i)}{\partial\tilde{y}_i}$ can be calculated with

$$\frac{\partial\mathcal{L}_d(\tilde{y}_i)}{\partial\tilde{y}_i} = \sum_{j=1}^{\tau}\left[\mathcal{F}'_j(\hat{y}_{ij})\prod_{m\neq j}^{\tau}\mathcal{F}_m(\hat{y}_{ij})\right], \tag{12}$$

The derivative function $\mathcal{F}'_j(\hat{y}_{ij})$ in Eq. (12) can be computed with

$$\mathcal{F}'_j(\hat{y}_{ij}) = \mathcal{H}'(\hat{y}_{ij})\mathcal{E}(\hat{y}_{ij}) + \mathcal{E}'(\hat{y}_{ij})\mathcal{H}(\hat{y}_{ij}), \tag{13}$$

In Eq. 13, $\mathcal{H}'(\hat{y}_{ij})$ and $\mathcal{E}'(\hat{y}_{ij})$ can be calculated by

$$\begin{aligned}\frac{\partial\mathcal{H}(\hat{y}_{ij})}{\partial\hat{y}_{ij}} &= \frac{1}{\sqrt{2\pi}}\left(-\frac{1}{2}\varphi_i^{-\frac{3}{2}}\right)\varphi'_{ij}\\ &= -\frac{\varphi'_{ij}}{2\varphi_i}\mathcal{H}(\hat{y}_{ij}),\\ \frac{\partial\mathcal{E}(\hat{y}_{ij})}{\partial\hat{y}_{ij}} &= \left[\frac{-\hat{y}_{ij}^2}{2\varphi_i}\right]'\mathcal{E}(\hat{y}_{ij})\\ &= \left[\frac{\hat{y}_{ij}^2\varphi'_{ij} - 2\hat{y}_{ij}\varphi_i}{2\varphi_i^2}\right]\mathcal{E}(\hat{y}_{ij}).\end{aligned} \tag{14}$$

Putting Eq. 14 into Eq. (13), $\mathcal{F}'_j(\hat{y}_{ij})$ can be rewritten as

$$\begin{aligned}\mathcal{F}'_j(\hat{y}_{ij}) &= \left[\frac{\hat{y}_{ij}^2\varphi'_{ij} - 2\hat{y}_{ij}\varphi_i}{2\varphi_i^2} - \frac{\varphi'_{ij}}{2\varphi_i}\right]\mathcal{F}_j(\hat{y}_{ij})\\ &= \left[\frac{\hat{y}_{ij}^2\varphi'_{ij} - 2\hat{y}_{ij}\varphi_i - \varphi_i\varphi'_{ij}}{2\varphi_i^2}\right]\mathcal{F}_j(\hat{y}_{ij}),\end{aligned} \tag{15}$$

Then, putting Eq. (15) into Eq. (12), $\frac{\partial\mathcal{L}_d(\tilde{y}_i)}{\partial\tilde{y}_i}$ can be rewritten as

$$\begin{aligned}\frac{\partial\mathcal{L}_d(\tilde{y}_i)}{\partial\tilde{y}_i} &= \sum_{j=1}^{\tau}\left[\mathcal{F}'_j(\hat{y}_{ij})\prod_{m\neq j}^{\tau}\mathcal{F}_m(\hat{y}_{ij})\right]\\ &= \sum_{j=1}^{\tau}\left[\left(\frac{\hat{y}_{ij}^2\varphi'_{ij} - 2\hat{y}_{ij}\varphi_i - \varphi_i\varphi'_{ij}}{2\varphi_i^2}\right)\prod_{j=1}^{\tau}\mathcal{F}_m(\hat{y}_{ij})\right]\\ &= \sum_{j=1}^{\tau}\left[\left(\frac{\hat{y}_{ij}^2\varphi'_{ij} - 2\hat{y}_{ij}\varphi_i - \varphi_i\varphi'_{ij}}{2\varphi_i^2}\right)\mathcal{L}_d(\tilde{y}_i)\right],\end{aligned} \tag{16}$$

where φ'_i is the partial derivative function φ_i with respect to $\hat{y}_{ij}$. We refer to Sen et al. (2005) and Guariglia (2021), φ'_i can be computed with

$$\frac{\partial\varphi_i}{\partial\hat{y}_{ij}} = -\frac{\varphi_i + \log(\hat{y}_{ij})}{1 - \hat{y}_{ij}}. \tag{17}$$

We also give the derivative function of entropy $\mathcal{L}'_e(p)$ for binary classification. It can be calculated by

$$\begin{aligned}\frac{\partial\mathcal{L}_e(p)}{\partial p} &= -\left[log(p) - log(1-p)\right]\\ &= log\left(\frac{1-p}{p}\right).\end{aligned} \tag{18}$$

Acknowledgements This work was supported by the National Natural Science Fund of China (62076184, 61976158, 61976160, 62076182, 62276190), in part by Fundamental Research Funds for the Central Universities and State Key Laboratory of Integrated Services Networks (Xidian University); and in part by Shanghai Innovation Action Project of Science and Technology (20511100700) and Shanghai Natural Science Foundation (22ZR1466700). Thanks to Xiaopeng Ji (Xiaopeng Ji is with the State Key Lab of CAD&CG, Zhejiang University, China. (email: xp.ji@cad.zju.edu.cn)) and Xinyang Jiang (Xinyang Jiang is with the Microsoft Research Asia (Shanghai), Shanghai, China. (email: xinyangjiang@microsoft.com)) for their help with this work.

Data Availibility The datasets used during and analyzed during the current study are available in the following public domain resources: https://image-net.org/index.php, https://www.cs.toronto.edu/~kriz/cifar.html, https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html, https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data, http://www.cbsr.ia.ac.cn/english/CASIA-WebFace-Database.html, https://www.robots.ox.ac.uk/$\sim$vgg/data/flowers/102/, http://host.robots.ox.ac.uk/pascal/VOC/, http://vis-www.cs.umass.edu/lfw/, http://whdeng.cn/CALFW/index.html, http://whdeng.cn/CPLFW/index.html, https://ibug.doc.ic.ac.uk/resources/agedb/, http://www.cfpw.io/, http://rose1.ntu.edu.sg/datasets/actionrecognition.asp, The models and source data generated during and analyzed during the current study are available from the corresponding author upon reasonable request.

References

Arbelaez, P., Maire, M., Fowlkes, C., & Malik, J. (2010). Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 33*(5), 898–916.

Arora, S., Ge, R., Neyshabur, B., & Zhang, Y. (2018). Stronger generalization bounds for deep nets via a compression approach. In *International Conference on Machine Learning*, PMLR, pp 254–263.

Banburski, A., De La Torre, F., Pant, N., Shastri, I., & Poggio, T (2021). Distribution of classification margins: Are all data equal? arXiv preprint arXiv:2107.10199

Cao, D., Zhu, X., Huang, X., Guo, J., & Lei, Z. (2020). Domain balancing: Face recognition on long-tailed domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp 5671–5679.

Castellano, G., & Vessio, G. (2022). A deep learning approach to clustering visual arts. *International Journal of Computer Vision, 130*(11), 2590–2605.

De Boer, P. T., Kroese, D. P., Mannor, S., & Rubinstein, R. Y. (2005). A tutorial on the cross-entropy method. *Annals of Operations Research, 134*, 19–67.

Deng, J., Guo, J., Xue, N., & Zafeiriou, S. (2019). Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp 4690–4699.

DeVries, T., & Taylor, G. W. (2017). Improved regularization of convolutional neural networks with cutout. arXiv preprint arXiv:1708.04552

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., & Uszkoreit, J. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. ICLR.

Du, B., Ye, J., Zhang, J., Liu, J., & Tao, D. (2022). I3cl: Intra-and inter-instance collaborative learning for arbitrary-shaped scene text detection. *International Journal of Computer Vision, 130*(8), 1961–1977.

Dynkin, E. (1978). Sufficient statistics and extreme points. *The Annals of Probability, 6*(5), 705–730.

Everingham, M., Eslami, S. A., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision, 111*, 98–136.

Gong, C., Liu, T., Tang, Y., Yang, J., Yang, J., & Tao, D. (2017). A regularization approach for instance-based superset label learning. *IEEE Transactions on Cybernetics, 48*(3), 967–978.

Goodfellow, I. J., Erhan, D., Carrier, P. L., Courville, A., Mirza, M., Hamner, B., Cukierski, W., Tang, Y., Thaler, D., Lee, D. H., & Zhou, Y. (2013). Challenges in representation learning: A report on three machine learning contests. In: International conference on neural information processing, Springer, pp 117–124.

Guariglia, E. (2021). Fractional calculus, zeta functions and Shannon entropy. *Open Mathematics, 19*(1), 87–100.

Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. In *International Conference on Machine Learning*, PMLR, pp 1321–1330.

Guo, J., Zhu, X., Zhao, C., Cao, D., Lei, Z., & Li, S. Z. (2020). Learning meta face recognition in unseen domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp 6163–6172

Hadsell, R., Chopra, S., & LeCun, Y. (2006). Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, IEEE, pp 1735–1742.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In: CVPR, pp 770–778.

Huang, G. B., Mattar, M., Berg, T., & Learned-Miller, E. (2008). Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*.

Ji, X., Zhao, Q., Cheng, J., & Ma, C. (2021). Exploiting spatio-temporal representation for 3d human action recognition from depth map sequences. *Knowledge-Based Systems, 227*, 107040.

Kanezaki, A. (2018). Unsupervised image segmentation by backpropagation. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, pp 1543–1547.

Khaireddin, Y., & Chen, Z. (2021). Facial emotion recognition: State of the art performance on fer2013. arXiv preprint arXiv:2105.03588

Kim, W., Kanezaki, A., & Tanaka, M. (2020). Unsupervised learning of image segmentation based on differentiable feature clustering. *IEEE Transactions on Image Processing, 29*, 8055–8068.

Kong, Y., & Fu, Y. (2022). Human action recognition and prediction: A survey. *International Journal of Computer Vision, 130*(5), 1366–1401.

Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. Master's thesis, University of Tront.

Krizhevsky, A. (2014). One weird trick for parallelizing convolutional neural networks. arXiv preprint arXiv:1404.5997

Lee, C. Y., Xie, S., Gallagher, P., Zhang, Z., & Tu, Z. (2015). Deeply-supervised nets. In *Artificial intelligence and statistics*, PMLR, pp 562–570.

Li, H., Jiang, T., & Zhang, K. (2003). Efficient and robust feature extraction by maximum margin criterion. Advances in neural information processing systems. Vol. *16*.

Liu, W., Wen, Y., Yu, Z., & Yang, M. (2016). Large-margin softmax loss for convolutional neural networks. In *ICML*, p 7.

Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., & Song, L. (2017). Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 212–220.

Moschoglou, S., Papaioannou, A., Sagonas, C., Deng, J., Kotsia, I., & Zafeiriou, S. (2017). Agedb: The first manually collected, in-the-wild age database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp 51–59.

Müller, R., Kornblith, S., & Hinton, G. (2019). When does label smoothing help? arXiv preprint arXiv:1906.02629

Nilsback, M. E., & Zisserman, A. (2008). Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*.

Qian, N. (1999). On the momentum term in gradient descent learning algorithms. *Neural Networks, 12*(1), 145–151.

Radosavovic, I., Kosaraju, R. P., Girshick, R., He, K., & Dollár, P. (2020). Designing network design spaces. In: CVPR.

Russakovsky, O., Deng, J., Su, H., et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision, 115*(3), 211–252.

Schroff, F., Kalenichenko, D., Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 815–823.

Sen, K., De Proft, F., Borgoo, A., et al. (2005). N-derivative of shannon entropy of shape function for atoms. *Chemical Physics Letters, 410*(1–3), 70–76.

Sengupta, S., Chen, J. C., Castillo, C., Patel, V. M., Chellappa, R., & Jacobs, D. W. (2016). Frontal to profile face verification in the wild. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, pp 1–9.

Sengupta, S., Chen, J. C., Castillo, C., Patel, V. M., Chellappa, R., & Jacobs, D. W. (2016). Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *CVPR*, pp 1010–1019.

Shi, Y., & Jain, A. K. (2019). Probabilistic face embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Sun, Y., Cheng, C., Zhang, Y., Zhang, C., Zheng, L., Wang, Z., & Wei, Y. (2020). Circle loss: A unified perspective of pair similarity optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp 6398–6407.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z., (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 2818–2826

Toneva, M., Sordoni, A., Combes, R. T. D., Trischler, A., Bengio, Y., & Gordon, G. J. (2019). An empirical study of example forgetting during deep neural network learning. In *ICLR*.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., & Jégou, H. (2021). Training data-efficient image transformers and distillation through attention. In *International Conference on Machine Learning*, pp 10,347–10,357.

Trockman, A., & Kolter, J. Z. (2022). Patches are all you need? arXiv preprint arXiv:2201.09792

Wang, D. B., Zhang, M. L., Li, L. (2021). Adaptive graph guided disambiguation for partial label learning. IEEE Transactions on Pattern Analysis and Machine Intelligence.

Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., & Liu, W. (2018). Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 5265–5274.

Wang, Q. W., Li, Y. F., Zhou, Z. H. (2019a). Partial label learning with unlabeled data. In *IJCAI*, pp 3755–3761.

Wang, Y., Ma, X., Chen, Z., Luo, Y., Yi, J., & Bailey, J. (2019b). Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp 322–330.

Wu, B., Jia, F., Liu, W., et al. (2018). Multi-label learning with missing labels using mixed dependency graphs. *International Journal of Computer Vision, 126*(8), 875–896.

Xu, N., Qiao, C., Geng, X., et al. (2021). Instance-dependent partial label learning. *Advances in Neural Information Processing Systems, 34*, 27,119-27,130.

Xu, X., Meng, Q., Qin, Y., Guo, J., Zhao, C., Zhou, F., & Lei, Z. (2021b). Searching for alignment in face recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp 3065–3073.

Yao, Y., Deng, J., Chen, X., Gong, C., Wu, J., & Yang, J. (2020). Deep discriminative CNN with temporal ensembling for ambiguously-labeled image classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp 12,669–12,676.

Yi, D., Lei, Z., Liao, S., & Li, S. Z. (2014). Learning face representation from scratch. arXiv preprint arXiv:1411.7923

Yin, D., Pananjady, A., Lam, M., Papailiopoulos, D., Ramchandran, K., & Bartlett, P., (2018). Gradient diversity: A key ingredient for scalable distributed learning. In *International Conference on Artificial Intelligence and Statistics, PMLR*, pp 1998–2007.

Yuan, L., Tay, F. E., Li, G., Wang, T., & Feng, J. (2020). Revisiting knowledge distillation via label smoothing regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp 3903–3911.

Zhang, C. B., Jiang, P. T., Hou, Q., et al. (2021). Delving deep into label smoothing. *IEEE Transactions on Image Processing, 30*, 5984–5996.

Zhang, L., Song, J., Gao, A., Chen, J., Bao, C., & Ma, K. (2019). Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp 3713–3722.

Zhao, Q., Wang, Y., Dou, S., Gong, C., Wang, Y., & Zhao, C. (2022). Adaptive discriminative regularization for visual classification. arXiv preprint arXiv:2203.00833

Zheng, T., & Deng, W. (2018). Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments. *Beijing University of Posts and Telecommunications, Tech Rep, 5*, 7.

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

Zhu, X., Liu, H., Lei, Z., et al. (2019). Large-scale bisample learning on id versus spot face recognition. *International Journal of Computer Vision, 127*(6), 684–700.

Zidek, J. V., & van Eeden, C. (2003). Uncertainty, entropy, variance and the effect of partial information. Lecture Notes-Monograph Series pp 155–167.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.