Twin Contrastive Learning with Noisy Labels

Zhizhong Huang[1]　　　Junping Zhang[1]　　　Hongming Shan[2,3]*

[1] Shanghai Key Lab of Intelligent Information Processing, School of Computer Science,
Fudan University, Shanghai 200433, China

[2] Institute of Science and Technology for Brain-inspired Intelligence and MOE Frontiers Center
for Brain Science, Fudan University, Shanghai 200433, China

[3] Shanghai Center for Brain Science and Brain-inspired Technology, Shanghai 200031, China

{zzhuang19, jpzhang, hmshan}@fudan.edu.cn

Abstract

Learning from noisy data is a challenging task that significantly degenerates the model performance. In this paper, we present TCL, a novel twin contrastive learning model to learn robust representations and handle noisy labels for classification. Specifically, we construct a Gaussian mixture model (GMM) over the representations by injecting the supervised model predictions into GMM to link label-free latent variables in GMM with label-noisy annotations. Then, TCL detects the examples with wrong labels as the out-of-distribution examples by another two-component GMM, taking into account the data distribution. We further propose a cross-supervision with an entropy regularization loss that bootstraps the true targets from model predictions to handle the noisy labels. As a result, TCL can learn discriminative representations aligned with estimated labels through mixup and contrastive learning. Extensive experimental results on several standard benchmarks and real-world datasets demonstrate the superior performance of TCL. In particular, TCL achieves 7.5% improvements on CIFAR-10 with 90% noisy label—an extremely noisy scenario. The source code is available at https://github.com/Hzzone/TCL.

1. Introduction

Deep neural networks have shown exciting performance for classification tasks [13]. Their success largely results from the large-scale curated datasets with clean human annotations, such as CIFAR-10 [19] and ImageNet [6], in which the annotation process, however, is tedious and cumbersome. In contrast, one can easily obtain datasets with some noisy annotations—from online shopping websites [40], crowd-sourcing [42, 45], or Wikipedia [32]—for training a clas-

sification neural network. Unfortunately, the mislabelled data are prone to significantly degrade the performance of deep neural networks. Therefore, there is considerable interest in training noise-robust classification networks in recent years [20, 21, 25, 29, 31, 48].

To mitigate the influence of noisy labels, most of the methods in literature propose the robust loss functions [37, 47], reduce the weights of noisy labels [35, 39], or correct the noisy labels [20, 29, 31]. In particular, label correction methods have shown great potential for better performance on the dataset with a high noise ratio. Typically, they correct the labels by using the combination of noisy labels and model predictions [31], which usually require an essential iterative sample selection process [1, 20, 21, 29]. For example, Arazo *et al.* [1] uses the small-loss trick to carry out sample selection and correct labels via the weighted combination. In recent years, contrastive learning has shown promising results in handling noisy labels [21, 21, 29]. They usually leverage contrastive learning to learn discriminative representations, and then clean the labels [21, 29] or construct the positive pairs by introducing the information of nearest neighbors in the embedding space. However, using the nearest neighbors only considers the label noise within a small neighborhood, which is sub-optimal and cannot handle extreme label noise scenarios, as the neighboring examples may also be mislabeled at the same time.

To address this issue, this paper presents TCL, a novel twin contrastive learning model that explores the *label-free* unsupervised representations and *label-noisy* annotations for learning from noisy labels. Specifically, we leverage contrastive learning to learn discriminative image representations in an unsupervised manner and construct a Gaussian mixture model (GMM) over its representations. Unlike unsupervised GMM, TCL links the *label-free* GMM and *label-noisy* annotations by replacing the latent variable of GMM with the model predictions for updating the parameters of GMM. Then, benefitting from the learned data

distribution, we propose to formulate label noise detection as an out-of-distribution (OOD) problem, utilizing another two-component GMM to model the samples with clean and wrong labels. The merit of the proposed OOD label noise detection is to take the full data distribution into account, which is robust to the neighborhood with strong label noise. Furthermore, we propose a bootstrap cross-supervision with an entropy regulation loss to reduce the impact of wrong labels, in which the true labels of the samples with wrong labels are estimated from another data augmentation. Last, to further learn robust representations, we leverage contrastive learning and Mixup techniques to inject the structural knowledge of classes into the embedding space, which helps align the representations with estimated labels.

The contributions are summarized as follows:

- We present TCL, a novel twin contrastive learning model that explores the *label-free* GMM for unsupervised representations and *label-noisy* annotations for learning from noisy labels.

- We propose a novel OOD label noise detection method by modeling the data distribution, which excels at handling extremely noisy scenarios.

- We propose an effective cross-supervision, which can bootstrap the true targets with an entropy loss to regularize the model.

- Experimental results on several benchmark datasets and real-world datasets demonstrate that our method outperforms the existing state-of-the-art methods by a significant margin. In particular, we achieve 7.5% improvements in extremely noisy scenarios.

2. Related Work

Contrastive learning. Contrastive learning methods [3, 12, 38] have shown promising results for both representation learning and downstream tasks. The popular loss function is InfoNCE loss [28], which can pull together the data augmentations from the same example and push away the other negative examples. MoCo [12] uses a memory queue to store the consistent representations. SimCLR [3] optimizes InfoNCE within mini-batch and has found some effective training tricks, *e.g.*, data augmentation. However, as unsupervised learning, they mainly focus on inducing transferable representations for the downstream tasks instead of training with noisy annotations. Although supervised contrastive learning [17] can improve the representations by human labels, it harms the performance when label noise exists [23].

Learning with noisy labels. Most of the methods in literature mitigate the label noise by robust loss functions [7,25,36, 37,41,47], noise transition matrix [8,30,35,39], sample selection [11,44], and label correction [18,20–22,25,29,31,34].

In particular, label correction methods have shown promising results than other methods. Arazo *et al.* [1] applied a mixture model to the losses of each sample to distinguish the noisy and clean labels, inspired by the fact that the noisy samples have a higher loss during the early epochs of training. Similarly, DivideMix [20] employs two networks to perform the sample selection for each other and applies the semi-supervised learning technique where the targets are computed from the average predictions of different data augmentations. Due to the success of contrastive learning, many attempts have been made to improve the robustness of classification tasks by combining the advantages of contrastive learning. Zheltonozhskii *et al.* [48] used contrastive learning to pre-train the classification model. MOIT [29] quantifies this agreement between feature representation and original label to identify mislabeled samples by utilizing a k-nearest neighbor (k-NN) search. RRL [21] performs label cleaning by two thresholds on the soft label, which is calculated from the predictions of previous epochs and its nearest neighbors. Sel-CL [23] leverages the nearest neighbors to select confident pairs for supervised contrastive learning [17].

Unlike existing methods [21,23,29] that detect the wrong labels within the neighborhood, TCL formulates the wrong labels as the out-of-distribution examples by modeling the data distribution of representations learned by contrastive learning. In addition, we propose a cross-supervision with entropy regularization to better estimate the true labels and handle the noisy labels.

3. The Proposed TCL

Each image in dataset $\mathcal{D} = \{\boldsymbol{x}_i\}_{i=1}^{N}$ associates with an annotation $y \in \{1, 2, \ldots, K\}$. In practice, some examples may be mislabeled. We aim to train a classification network, $p_\theta(y|\boldsymbol{x}) = g(\boldsymbol{x}; \theta) \in \mathbb{R}^K$, that is resistant to the noisy labels in training data, and generalizes well on the clean testing data. Fig. 1 illustrates the framework of our proposed TCL.

Overview. In the context of our framework, $f(\cdot)$ and $g(\cdot)$ share the same backbone and have additional individual heads to output representations and class predictions from two random and one mixup data augmentations. Afterward, there are four components in TCL, including (i) modeling the data distribution via a GMM in Sec. 3.1 from the model predictions and representations; (ii) detecting the examples with wrong labels as out-of-distribution samples in Sec. 3.2; (iii) cross-supervision by bootstrapping the true targets in Sec. 3.3; and (iv) learning robust representations through contrastive learning and mixup in Sec. 3.4.

3.1. Modeling Data Distribution

Given the image dataset consisting of N images, we opt to model the distribution of $\boldsymbol{x}$ over its representation $\boldsymbol{v} = f(\boldsymbol{x})$ via a spherical Gaussian mixture model (GMM). After in-
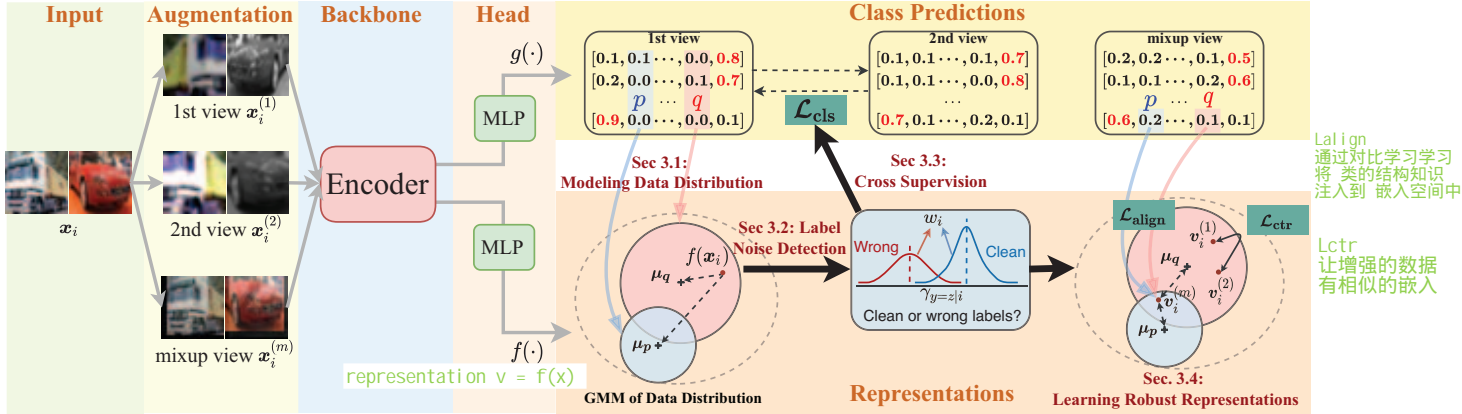
Figure 1. Illustration of the proposed TCL. The networks g and f with shared encoder and independent two-layer MLP output the class predictions and representations. Then, TCL models the data distribution via a GMM, and detects the examples with wrong labels as out-of-distribution examples. To optimize TCL, these results lead to cross-supervision and robust representation learning.

troducing discrete latent variables $z \in \{1, 2, \ldots, K\}$ that determine the assignment of observations to mixture components, the unsupervised GMM can be defined as

$$p(\boldsymbol{v}) = \sum_{k=1}^{K} p(\boldsymbol{v}, z = k)$$
$$= \sum_{k=1}^{K} p(z = k) \mathcal{N}(\boldsymbol{v} | \boldsymbol{\mu}_k, \sigma_k). \quad (1)$$

where $\boldsymbol{\mu}_k$ is the mean and σ_k a scalar deviation. If we assume that the latent variables z are uniform distributed, that is, $p(z = k) = 1/K$, we can define the posterior probability that assigns $\boldsymbol{x}_i$ to k-th cluster:

$$\gamma_{ik} = p(z_i = k | \boldsymbol{x}_i) \propto \mathcal{N}(\boldsymbol{x}_i | \boldsymbol{\mu}_k, \sigma_k). \quad (2)$$

In an ideal scenario where all the samples have clean labels $y \in \{1, 2, \ldots, K\}$, the discrete latent variables z would be identical to the annotation y, and the parameters $\boldsymbol{\mu}_k$, σ_k and latent variable z can be solved through a standard Expectation-Maximization (EM) algorithm [5].

However, in practice, the labels are often noisy and the latent variable z, estimated in an unsupervised manner, has nothing to do with the label y. Therefore, we are interested in connecting latent variable z, estimated *in an unsupervised fashion (*i.e. *label-free)*, and the available annotations y, *label-noisy*, for the task of learning from noisy labels.

To link them together, we propose to inject the model predictions $p_\theta(y_i = k | \boldsymbol{x}_i)$, learned from noisy labels, into the latent variables z. Specifically, we propose to replace the unsupervised assignment $p(z_i = k | \boldsymbol{x}_i)$ with noisy-supervised assignment $p_\theta(y_i = k | \boldsymbol{x}_i)$. As a result, we can connect the latent variable z with the label y, and thus use the noisy supervision to guide the update of the parameters of GMM.

In particular, the update of the GMM parameters becomes

$$\boldsymbol{\mu}_k = \text{norm} \left(\frac{\sum_i p_\theta(y_i = k | \boldsymbol{x}_i) \boldsymbol{v}_i}{\sum_i p_\theta(y_i = k | \boldsymbol{x}_i)} \right), \quad (3)$$

$$\sigma_k = \frac{\sum_i p_\theta(y_i = k | \boldsymbol{x}_i)(\boldsymbol{v}_i - \boldsymbol{\mu}_k)(\boldsymbol{v}_i - \boldsymbol{\mu}_k)^{\mathrm{T}}}{\sum_i p_\theta(y_i = k | \boldsymbol{x}_i)}, \quad (4)$$

where $\text{norm}(\cdot)$ is ℓ_2-normalization such that $\|\boldsymbol{\mu}_k\|_2 = 1$.

3.2. Out-Of-Distribution Label Noise Detection

Previous works [21, 23, 29] typically detect the wrong labels within the neighborhood, that is, using the information from nearest neighbors. It is limited as the neighboring examples are usually mislabeled at the same time. To address this issue, we propose to formulate label noise detection as to detect the out-of-distribution examples.

After building the connection between the latent variables z and labels y, we are able to detect the sample with wrong labels through the posterior probability in Eq. (2). We implement it as a normalized version to take into account the intra-cluster distance, which allows for detecting the samples with likely wrong labels:

$$\gamma_{ik} = \frac{\exp\left(-(\boldsymbol{v}_i - \boldsymbol{\mu}_k)^{\mathrm{T}}(\boldsymbol{v}_i - \mu_k)/2\sigma_k\right)}{\sum_k \exp\left(-(\boldsymbol{v}_i - \boldsymbol{\mu}_k)^{\mathrm{T}}(\boldsymbol{v}_i - \mu_k)/2\sigma_k\right)}. \quad (5)$$

Since ℓ_2-normalization has been applied to both embeddings $\boldsymbol{v}$ and the cluster centers $\boldsymbol{\mu}_k$, yielding $(\boldsymbol{v} - \boldsymbol{\mu}_k)^{\mathrm{T}}(\boldsymbol{v} - \boldsymbol{\mu}_k) = 2 - 2\boldsymbol{v}^{\mathrm{T}}\boldsymbol{\mu}_k$. Therefore, we can re-write Eq. (5) as:

$$\gamma_{ik} = p(z_i = k | \boldsymbol{x}_i)$$
$$= \exp(\boldsymbol{v}_i^{\mathrm{T}} \boldsymbol{\mu}_k / \sigma_k) \Big/ \sum_k \exp(\boldsymbol{v}_i^{\mathrm{T}} \boldsymbol{\mu}_k / \sigma_k). \quad (6)$$

Once built the GMM over the distribution of representations, we propose to formulate the conventional *noisy label*

detection problem as *out-of-distribution sample* detection problem. Our idea is that the samples with clean labels should have the same cluster indices after linking the cluster index and class label. Specifically, given one particular class $y = k$, the samples within this class can be divided into two types: in-distribution samples with clean labels, and out-of-distribution samples with wrong labels. Therefore, we define the following conditional probability to measure the probability of one sample with clean label:

$$\gamma_{y=z|i} = p(y_i = z_i|\boldsymbol{x}_i)$$
$$= \exp(\boldsymbol{v}_i^{\mathrm{T}}\boldsymbol{\mu}_{z_i}/\sigma_{z_i}) \Big/ \sum_k \exp(\boldsymbol{v}_i^{\mathrm{T}}\boldsymbol{\mu}_k/\sigma_k). \quad (7)$$

Although Eqs. (6) and (7) share similar calculations, they have different meanings. Eq. (6) calculates the probability of one example *belonging to k-th cluster* while Eq. (7) the probability of one example *having clean label*—that is, $y_i = z_i$. Therefore, the probability of one example having the wrong label can be written as $\gamma_{y \neq z|i} = p(y_i \neq z_i|\boldsymbol{x}_i) = 1 - p(y_i = z_i|\boldsymbol{x}_i)$.

Furthermore, instead of setting a human-tuned threshold for $\gamma_{y=z|i}$, we opt to employ another two-component GMM following [1, 20] to automatically estimate the clean probability $\gamma_{y=z|i}$ for each example. Similar to the definition of GMM in Eq. (1), this two-components GMM is defined as follows:

$$p(\gamma_{y=z|i}) = \sum_{c=0}^{1} p(\gamma_{y=z|i}, c) = \sum_{c=0}^{1} p(c)p(\gamma_{y=z|i}|c), \quad (8)$$

where c is the new introduced latent variable: $c = 1$ indicates the cluster of clean labels with higher mean value and vice versus $c = 0$. After modeling the GMM over the probability of one example having clean labels, $\gamma_{y=z|i}$, we are able to infer the posterior probability of one example having clean labels through the two-component GMM.

3.3. Cross-supervision with Entropy Regularization

After the label noise detection, the next important step is to estimate the true targets by correcting the wrong label to reduce its impact, called label correction. Previous works usually perform label correction using the temporal ensembling [25] or from the model predictions [1, 20] before mixup augmentation without back-propagation.

TCL leverages a similar idea to bootstrap the targets through the convex combination of its noisy labels and the predictions from the model itself:

$$\begin{cases} \boldsymbol{t}_i^{(1)} = w_i\boldsymbol{y}_i + (1 - w_i)g(\boldsymbol{x}_i^{(1)}) \\ \boldsymbol{t}_i^{(2)} = w_i\boldsymbol{y}_i + (1 - w_i)g(\boldsymbol{x}_i^{(2)}) \end{cases}, \quad (9)$$

where $g(\boldsymbol{x}_i^{(1)})$ and $g(\boldsymbol{x}_i^{(2)})$ are the predictions of two augmentations, $\boldsymbol{y}_i$ the noisy one-hot label, and $w_i \in [0, 1]$ represents the posterior probability as $p(c = 1|\gamma_{y=z|i})$ from the

two-component GMM defined in Eq. (8). When computing Eq. (9), we stop the gradient from g to avoid the model predictions collapsed into a constant, inspired by [4, 10].

Guided by the corrected labels $\boldsymbol{t}_i$, we swap two augmentations to compute the classification loss twice, leading to the bootstrap cross supervision, formulated as:

$$\mathcal{L}_{\text{cross}} = \ell\left(g(\boldsymbol{x}_i^{(1)}), \boldsymbol{t}_i^{(2)}\right) + \ell\left(g(\boldsymbol{x}_i^{(2)}), \boldsymbol{t}_i^{(1)}\right), \quad (10)$$

where ℓ is the cross-entropy loss. This loss makes the predictions of the model from two data augmentations close to corrected labels from each other. In a sense, if $w_i = 0$, the model is encouraged for consistent class predictions between different data augmentations, otherwise $w_i = 1$ it is supervised by the clean labels.

In addition, we leverage an additional entropy regularization loss on the predictions within a mini-batch $\mathcal{B}$:

$$\mathcal{L}_{\text{reg}} = -\mathbb{H}\left(\frac{1}{|\mathcal{B}|}\sum_{\boldsymbol{x}\in\mathcal{B}} g(\boldsymbol{x})\right) + \frac{1}{|\mathcal{B}|}\sum_{\boldsymbol{x}\in\mathcal{B}} \mathbb{H}\left(g(\boldsymbol{x})\right), \quad (11)$$

where $\mathbb{H}(\cdot)$ is the entropy of predictions [33]. The first term can avoid the predictions collapsing into a single class by maximizing the entropy of average predictions. The second term is the minimum entropy regularization to encourage the model to have high confidence for predictions, which was previously studied in semi-supervised learning literature [9].

Although both using the model predictions, we would emphasize that the cross-supervision in TCL is different to [1, 20, 25] in three aspects: (i) both $\boldsymbol{x}_i^{(1)}$ and $\boldsymbol{x}_i^{(2)}$ are involved in back-propagation; (ii) the strong augmentation [3] used to estimate the true targets can prevent the overfitting of estimated targets; and (iii) TCL employs two entropy regularization terms to avoid the model collapse to one class.

The final classification loss is given as follows:

$$\mathcal{L}_{\text{cls}} = \mathcal{L}_{\text{cross}} + \mathcal{L}_{\text{reg}}. \quad (12)$$

3.4. Learning Robust Representations

To model the data distribution that is robust to noisy labels, we leverage contrastive learning to learn the representations of images. Specifically, contrastive learning performs instance-wise discrimination [38] using the InfoNCE loss [28] to enforce the model outputting similar embeddings for the images with semantic preserving perturbations. Formally, the contrastive loss is defined as follows:

$$\mathcal{L}_{\text{ctr}} = -\log \frac{\exp\left(f(\boldsymbol{x}^{(1)})^{\mathrm{T}}f(\boldsymbol{x}^{(2)})/\tau\right)}{\sum_{\boldsymbol{x}\in\mathcal{S}} \exp\left(f(\boldsymbol{x}^{(1)})^{\mathrm{T}}f(\boldsymbol{x})/\tau\right)}, \quad (13)$$

where τ is the temperature and $\mathcal{S}$ is the $\mathcal{B}$ except $\boldsymbol{x}^{(1)}$. $\boldsymbol{x}^{(1)}$ and $\boldsymbol{x}^{(2)}$ are two augmentations of $\boldsymbol{x}$. Intuitively, InfoNCE loss aims to pull together the positive pair $(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)})$ from

two different augmentations of the same instance, and push them away from negative examples of other instances. Consequently, it can encourage discriminative representations in a pure unsupervised, or label-free manner.

Although beneficial in modeling latent representations, contrastive learning cannot introduce compact classes without using the true labels. Since the label y is noisy, we leverage Mixup [46] to improve within-class compactness, which has been shown its effectiveness against label noise in literature [1, 20]. Specifically, a mixup training pair $(\boldsymbol{x}_i^{(m)}, \bar{\boldsymbol{t}}_i^{(m)})$ is linearly interpolated between $(\boldsymbol{x}_i, \bar{\boldsymbol{t}}_i)$ and $(\boldsymbol{x}_j, \bar{\boldsymbol{t}}_j)$ under a control coefficient $\lambda \sim \text{Beta}(\alpha, \alpha)$:

$$\begin{cases} \boldsymbol{x}_i^{(m)} = \lambda \boldsymbol{x}_i + (1 - \lambda)\boldsymbol{x}_j, \\ \bar{\boldsymbol{t}}_i^{(m)} = \lambda \bar{\boldsymbol{t}}_i + (1 - \lambda)\bar{\boldsymbol{t}}_j, \end{cases} \quad (14)$$

where $\boldsymbol{x}_j$ is randomly selected within a mini-batch, and $\bar{\boldsymbol{t}}_i = (\boldsymbol{t}_i^{(1)} + \boldsymbol{t}_i^{(2)})/2$ is the average of estimated true labels of two data augmentations. Intuitively, we can inject the structural knowledge of classes into the embedding space learned by contrastive learning. This loss can be written as:

$$\mathcal{L}_{\text{align}} = \ell\left(g(\boldsymbol{x}_i^{(m)}), \bar{\boldsymbol{t}}_i^{(m)}\right) + \ell(p(\boldsymbol{z}|\boldsymbol{x}_i^{(m)}), \bar{\boldsymbol{t}}_i^{(m)}), \quad (15)$$

where the second term can align the representations with estimated labels. In a sense, $\mathcal{L}_{\text{align}}$ regularizes classification network g and encourages f to learn compact and well-separated representations. Furthermore, we would point out two differences between TCL and [21], although both using mixup to boost the representations. First, [21] does not explicitly model the data distribution $p(\boldsymbol{z}|\boldsymbol{x}_i^{(m)})$ like TCL. Second, TCL has leveraged the full training dataset via the corrected label $\bar{\boldsymbol{t}}_i^{(m)}$ instead of a subset of clean examples in [21], which leads to stronger robustness of TCL over [21] on extreme high label noise ratios.

3.5. Training and inference

The overall training objective is to minimize the sum of all losses:

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{ctr}} + \mathcal{L}_{\text{align}}. \quad (16)$$

We find that a simple summation of all losses works well for all datasets and noise levels, which indicates the strong generalization of the proposed method. During inference, the data augmentations are disabled and the class predictions are obtained by $\arg\max_k p_\theta(k|\mathbf{x})$.

The training algorithm of the proposed method is shown in Alg. 1. In a sense, the architecture of our method leads to an EM-like algorithm: (1) the **E-step** updates $\{(\boldsymbol{\mu}_k, \sigma_k)\}_{k=1}^K$ for TCL, and $\{w_i\}_{i=1}^N$ for each sample in $\mathcal{D}$ to form the true targets with the predictions from another data augmentations, and (2) the **M-step** optimizes the model parameters by Eq. (16) to better fit those estimated targets. Therefore, the convergence of TCL can be theoretically guaranteed, following the standard EM algorithm.

Algorithm 1: Training Algorithm

Input: Dataset $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$; functions $\{f, g\}$
Output: Classification network g.
repeat
  E-step: update $\{(\boldsymbol{\mu}_k, \sigma_k)\}_{k=1}^K$ for TCL, and
  $\{w_i\}_{i=1}^N$ for each sample in $\mathcal{D}$
  M-step: **repeat**
    Randomly sample a mini-batch $\mathcal{B}$ from $\mathcal{D}$
    for each $\boldsymbol{x}_i$ in $\mathcal{B}$ **do**
      Randomly sample two augmentations
      and a mixup one: $\{\boldsymbol{x}_i^{(1)}, \boldsymbol{x}_i^{(2)}, \boldsymbol{x}_i^{(m)}\}$
      $\mathcal{L} \leftarrow$ Eq. (16)
    end
    Update f and g with SGD optimizer.
  until an epoch finished;
until reaching max epochs;

---

4. Experiments

In this section, we conduct experiments on multiple benchmark datasets with simulated and real-world label noises. We strictly follow the experimental settings in previous literature [20, 21, 25, 29] for fair comparisons.

4.1. Experiments on simulated datasets

Datasets. Following [20, 21, 25, 29], we validate our method on **CIFAR-10/100** [19], which contains 50K and 10K images with size 32×32 for training and testing, respectively. We leave 5K images from the training set as the validation set for hyperparameter tuning, then train the model on the full training set for fair comparisons. Two types of label noise are simulated: *symmetric* and *asymmetric* label noise. Symmetric noise randomly assigns the labels of the training set to random labels with predefined percentages, *a.k.a*, noise ratio, which includes 20%, 50%, 80%, and 90% on two datasets in this paper. Asymmetric noise takes into account the class semantic information, and the labels are only changed to similar classes (*e.g.*, truck → automobile). Here, only experiments on the CIFAR-10 dataset with 40% noise ratio for asymmetric noise are conducted; otherwise, the classes with above 50% label noise cannot be distinguished.

Training details. Same as previous works [20, 21, 25, 29], we use a PreAct ResNet-18 [14] as the encoder. We adopt SGD optimizer to train our model with a momentum of 0.9, a weight decay of 0.001, and a batch size of 256 for 200 epochs. The learning rate are linearly warmed up to 0.03 for 20 epochs and decayed with the cosine schedule. The data augmentations of [3] are applied to two views (ResizedCrop, ColorJitter, and *etc*). Only crop and horizontal flip are employed for mixup. Both projection and classification heads are a two-layer MLP with the dimension 128 and the number

11665

| Noise type/rate | CIFAR-10 | | | | | | CIFAR-100 | | | | |
| | Sym. | | | | Asym. | Avg. | Sym. | | | | Avg. |
	20%	50%	80%	90%	40%		20%	50%	80%	90%	
Cross-Entropy	82.7	57.9	26.1	16.8	76.0	51.9	61.8	37.3	8.8	3.5	27.8
Mixup (17') [46]	92.3	77.6	46.7	43.9	77.7	67.6	66.0	46.6	17.6	8.1	34.6
P-correction (19') [43]	92.0	88.7	76.5	58.2	91.6	81.4	68.1	56.4	20.7	8.8	38.5
M-correction (19') [1]	93.8	91.9	86.6	68.7	87.4	85.7	73.4	65.4	47.6	20.5	51.7
ELR (20') [25]	93.8	92.6	88.0	63.3	85.3	84.6	74.5	70.2	45.2	20.5	52.6
DivideMix (20') [20]	95.0	93.7	92.4	74.2	91.4	89.3	74.8	72.1	57.6	29.2	58.4
MOIT (21') [29]	93.1	90.0	79.0	69.6	92.0	84.7	73.0	64.6	46.5	36.0	55.0
RRL (21') [21]	**95.8**	**94.3**	92.4	75.0	91.9	89.8	**79.1**	**74.8**	57.7	29.3	60.2
Sel-CL+ (22') [23]	95.5	93.9	89.2	81.9	**93.4**	90.7	76.5	72.4	59.6	48.8	64.3
TCL (**ours**)	95.0	93.9	**92.5**	**89.4**	92.6	**92.7**	78.0	73.3	**65.0**	**54.5**	**67.7**
	±0.1	±0.1	±0.2	±0.2	±0.1		±0.2	±0.2	±0.3	±0.5	

Table 1. Comparisons with state-of-the-art methods on simulated datasets. The results for previous methods are copied from [21, 23] to avoid the bias of self-implementation, and we strictly follow their experimental settings. Each runs has been repeated 3 times with different randomly-generated noise and we report the mean and std values of *last* 5 epochs.
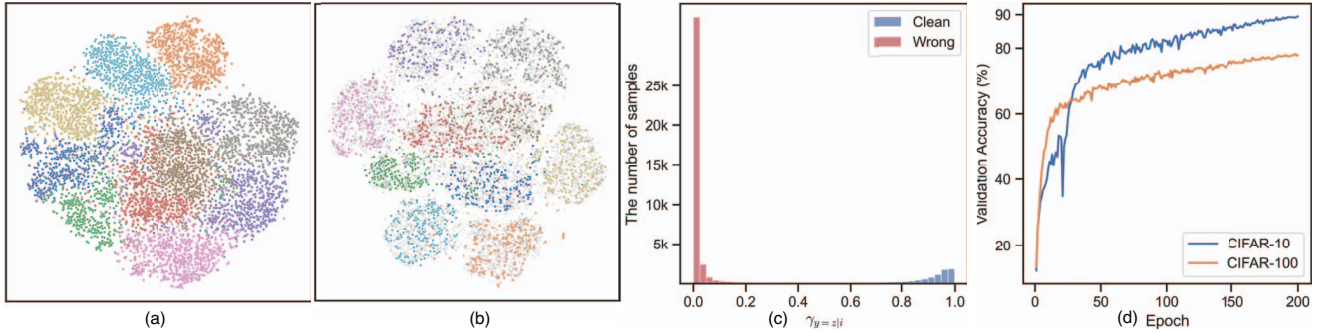

Figure 2. Qualitative results. For the model trained on CIFAR-10 with 90% *sym.* noise at 200th epoch, we show t-SNE visualizations for the learned representations of (a) testing set where different color denotes different class predicted by $g(\cdot)$ and (b) 10K samples from training set colored by the true labels; the gray '+' denotes the samples with noisy labels. (c) The histogram of $p(y = z|\boldsymbol{x})$ for full training set colored by the clean and noisy labels. (d) The validation accuracy across training of CIFAR-10 and CIFAR-100 on 90% *sym.* noise.

of classes. The temperature τ of contrastive loss and the α of mixup are 0.25 and 1. The settings are shared for all experiments, which are significantly different from [20, 21, 25] that adopt specific configurations for different datasets and even for different noise ratios/types.

Quantitative results. Table 1 presents the the comparisons with existing state-of-the-art methods. Our method yields competitive performance on low noise ratios, but promising improvements over recent methods on extreme noise ratios and the most challenging CIFAR-100 dataset with 100 classes. In particular, with 90% label noise, there are 7.5% and 5.7% improvements on for CIFAR-10 and CIFAR-100, respectively. We stress that the hyperparameters are consistent for different noise ratios/types. In practical scenarios, the noise ratio for a particular dataset is unknown, so it is hard to tune the hyper-parameters for better performance. Therefore, these results indicate the strong generalization ability of our method regardless of noise ratios/types.

For fair comparisons, following [23, 29], we performed extra experiments on fine-tuning the classification network for 70 epochs with the detected clean samples and mixup augmentation, termed TCL+. Table 2 shows that under low label noise (below 50%), TCL+ achieves significant improvements over TCL and outperforms the recent state-of-the-art methods. The benefits from the detected clean subset and longer training, which can fully utilize the useful supervision signals from labeled examples.

In Appendix A, we also perform the k-NN classification over the learned representations, which indicates that our method has maintained meaningful representations better than the pure unsupervised learning model. In Appendix B, we provide more experimental results and analysis on asymmetric label noise and imbalance data.

Qualitative results. Figs. 2(a) and (b) visualize the learned representations with extremely high noise ratio, demonstrating that our method can produce distinct structures of learned

	CIFAR-10			CIFAR-100	
	Sym.		*Asym.*	*Sym.*	
	20%	50%	40%	20%	50%
DivideMix [20]	95.0	93.7	91.4	74.8	72.1
MOIT [29]	93.1	90.0	92.0	73.0	64.6
MOIT+ [29]	94.1	91.8	93.3	75.9	70.6
RRL [21]	95.8	94.3	91.9	79.1	**74.8**
Sel-CL+ [23]	95.5	93.9	93.4	76.5	72.4
TCL (**ours**)	95.0	93.9	92.6	78.0	73.3
TCL+ (**ours**)	**96.0**	**94.5**	**93.7**	**79.3**	74.6

Table 2. Comparisons with SOTAs under *low* label noise.

representations with meaningful semantic information. Especially, Fig. 2(b) presents the samples with noisy labels in the embedding space, in which the label noise can be accurately detected by our proposed method. In addition, by visualizing the histogram of $p(y = z|\boldsymbol{x})$ for the training set in Fig. 2(c), we confirm that the proposed method can effectively distinguish the samples noisy and clean labels. We visualize the validation accuracy across training in Fig. 2(d). As expected, TCL performs stable even with the extreme 90% label noise.

4.2. Ablation study

We conduct ablation studies to validate our motivation and design with the following baselines, and the results are shown in Table 3.

(i) **Baseline.** We start the baseline method by removing the proposed noisy label detection and bootstrap cross-supervision, where the model is directly guided by noisy labels. As expected, the performance significantly degrades for the extremely high noise ratio (*i.e.*, 90%).

(ii) **Label Noise Detection.** We assess the effectiveness of different detection methods including the cross-entropy loss [1, 20], k-NN search [29], and our out-of-distribution (OOD) detection. For fair comparisons, the predictions from the images before mixup are employed as the true labels in Eq. (9). Obviously, the label noise detection has alleviated the degeneration to some degree (Exp. (i)), where our method consistently outperforms other baselines. Fig. 3 visualizes their AUCs across training. The proposed OOD detection is better at distinguishing clean and wrong labels. Thanks to the representations learned by contrastive learning, k-NN search performs better than cross-entropy loss. However, it is limited due to the use of the original labels to detect noisy ones, while our method constructs a GMM using the model predictions.

(iii) **Target Estimation.** Another key component is the cross-supervision that bootstraps the true targets from the predictions of another data augmentation. We replace it with the temporal ensembling [25], where the hyperparameters are set as suggested by [25]. Furthermore, Exp. (iii) estimates true targets from the images before mixup [1, 20, 25].
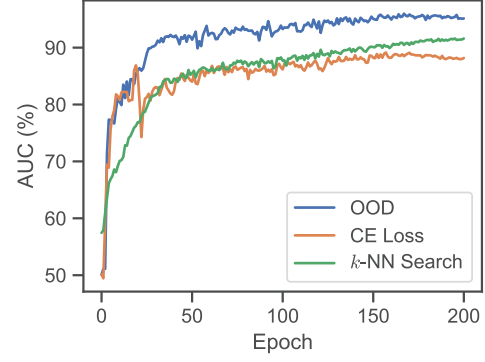

Figure 3. Training curve of AUC for noisy label detection trained on CIFAR-10 with 90% *sym.* noise.

Dataset		CIFAR-10				CIFAR-100		
		Sym.		*Asym.*	*Avg.*	*Sym.*		*Avg.*
Noise type/rate		50%	90%	40%		50%	90%	
(i)	Baseline	70.0	20.6	77.5	56.1	47.3	6.8	27.1
(ii)	Loss [1, 20]	92.5	75.9	73.2	80.6	71.2	16.0	43.6
	k-NN [29]	92.9	79.7	91.3	88.0	70.3	39.8	55.1
	OOD (**ours**)	93.1	82.1	92.0	89.1	70.7	45.9	58.3
(iii)	*Ensem.* [25]	91.3	72.7	89.8	84.6	68.2	36.9	52.6
	$\mathcal{L}_{\text{cross}}$ (**ours**)	93.9	89.4	92.6	92.0	73.3	54.5	63.9
(iv)	w/o $\mathcal{L}_{\text{reg}}$	92.0	34.5	90.3	72.3	68.5	24.3	46.4
(v)	w/o $\mathcal{L}_{\text{align}}$	91.8	84.6	89.7	88.7	69.4	48.4	58.9
(vi)	MoCo	94.4	90.7	93.1	92.7	74.0	57.3	65.6

Table 3. Ablation results of different components in TCL.

The results suggest that our bootstrap cross-supervision has shown strong robustness on 90% label noise.

(iv) **Without $\mathcal{L}_{\text{reg}}$.** We remove $\mathcal{L}_{\text{reg}}$ and the results indicate that it plays an important role, especially on extremely high label noise. Removing each term in $\mathcal{L}_{\text{reg}}$ obtains similar results. We argue that $\mathcal{L}_{\text{reg}}$ works in two aspects: 1) it can avoid the model collapse which outputs single classes, and 2) it can encourage the model to have high confidence for the predictions, which has shown its effectiveness for unlabeled data in semi-supervised learning.

(v) **Without $\mathcal{L}_{\text{align}}$.** We remove $\mathcal{L}_{\text{align}}$ and the performance has decreased, as expected, but is still more promising than other baselines. $\mathcal{L}_{\text{align}}$ has leveraged mixup augmentation to regularize both classification and representation learning. Appendix A shows the evaluation of k-NN classification, demonstrating that $\mathcal{L}_{\text{align}}$ can also greatly improve the learned representations.

(vi) **Contrastive Framework.** We implement TCL into another contrastive framework for representation learning, *i.e.*, MoCo [12]. Based on the MoCo framework, our method has achieved more improvements in various experiments, which benefits from a large number of negative examples in the memory queue and a moving-averaged encoder (we set the queue size and the factor of moving-average to 4,096 and 0.99, respectively).
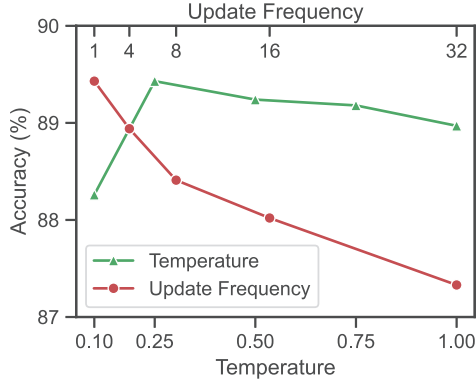
11667

Figure 4. Ablation results for hyperparameters.

	WebVision		**ILSVRC12**	
	top1	top5	top1	top5
Forward [30]	61.1	82.6	57.3	82.3
D2L [26]	62.6	84.0	57.8	81.3
Iterative-CV [2]	65.2	85.3	61.6	84.9
Decoupling [27]	62.5	84.7	58.2	82.2
MentorNet [16]	63.0	81.4	57.8	79.9
Co-teaching [11]	63.5	85.2	61.4	84.7
ELR [25]	76.2	91.2	68.7	87.8
DivideMix [20]	77.3	91.6	75.2	90.8
RRL [21]	76.3	91.5	73.3	91.2
NGC [22]	**79.1**	91.8	74.4	91.0
MOIT [29]	77.9	91.9	73.8	91.7
TCL (**ours**)	**79.1**	**92.3**	**75.4**	**92.4**

Table 4. Results on WebVision (mini).

Method	Acc (%)
Cross-Entropy	69.2
Label Correction [1]	71.0
Joint-Opt [34]	72.2
ELR [25]	72.8
SL [37]	74.4
DivideMix [20]	74.4
MentorMix [15]	74.3
RRL [21]	**74.8**
TCL (**ours**)	**74.8**

Table 5. Results on Clothing1M.

Hyperparameters. We evaluate the most essential hyperparameters to our design, including the temperature τ for contrastive loss and update frequency for TCL on CIFAR-10 with 90% symmetric noise. Here, the update frequency denotes how may epochs that we update the parameters of TCL, $\{(\boldsymbol{\mu}_k, \sigma_k)\}_{k=1}^{K}$ and $\{w_i\}_{i=1}^{N}$. Fig. 4 shows that our method is robust to different choices of hyperparameters. Even though TCL updates for every 32 epochs, our method has still performed well, which indicates that the computational cost can be significantly reduced.

4.3. Results on real-world datasets

Datasets and training details. We validate our method on two real-word noisy datasets: WebVision [24] and Clothing1M [40]. Webvision contains millions of noisily-labeled images collected from the web using the keywords of ImageNet ILSVRC12 [6]. Following the convension [20, 21, 25, 29], we conducted experiments on the first 50 classes of the Google image subset, termed WebVision (mini) and evaluated on both WebVision and ImageNet validation set. Clothing1M consists of 14 kinds of images collected from online shopping websites. Only the noisy training set is used in our experiments. We used a batch size of 256 on 4 GPUs, and trained a ResNet-50 for 40 epochs (without warm-up) on Clothing1M and a ResNet-18 for 130 epochs (warm-up 10 epochs) on WebVision, respectively. Following [20, 21, 25], for Clothing1M, the encoder is initialized with ImageNet pre-trained weights, the initial learning rate is 0.01, and 256 mini-batches are sampled as one epoch. Other hyper-parameters are kept to be the same without further tuning.

Quantitative results. Tables 4 and 5 present the results on WebVision and Clothing1M datasets. Our method outperforms state-of-the-art methods on both datasets, demonstrating its superior performance in handling real-world noisy datasets. We note that after checking the Clothing1M dataset, there are still lots of mislabeled images in the testing set.

Therefore, the results on Clothing1M may not be such reliable as other datasets to evaluate the true performance of different methods.

5. Conclusion

In this paper, we introduced TCL, a novel twin contrastive learning model for learning from noisy labels. By connecting the *label-free* latent variables and *label-noisy* annotations, TCL can effectively detect the label noise and accurately estimate the true labels. Extensive experiments on both simulated and real-world datasets have demonstrated the superior performance of TCL than existing state-of-the-art methods. In particular, TCL achieves 7.5% performance improvement under extremely 90% noise ratio. In the future, we will improve TCL with semantic information for low noise ratios and explore dynamically updating the GMM.

References

[1] Eric Arazo, Diego Ortego, Paul Albert, Noel O'Connor, and Kevin McGuinness. Unsupervised label noise modeling and loss correction. In *International Conference on Machine Learning*, pages 312–321. PMLR, 2019.

[2] Pengfei Chen, Ben Ben Liao, Guangyong Chen, and Shengyu Zhang. Understanding and utilizing deep neural networks trained with noisy labels. In *International Conference on Machine Learning*, pages 1062–1070. PMLR, 2019.

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020.

[4] Xinlei Chen and Kaiming He. Exploring simple Siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021.

[5] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. Ieee, 2009.

[7] Aritra Ghosh, Himanshu Kumar, and PS Sastry. Robust loss functions under label noise for deep neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2017.

[8] Jacob Goldberger and Ehud Ben-Reuven. Training deep neural-networks using a noise adaptation layer. In *International Conference on Learning Representations*, 2017.

[9] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in Neural Information Processing Systems*, 2004.

[10] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, 2020.

[11] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[12] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pages 630–645. Springer, 2016.

[15] Lu Jiang, Di Huang, Mason Liu, and Weilong Yang. Beyond synthetic noise: Deep learning on controlled noisy labels. In *International Conference on Machine Learning*, 2020.

[16] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. MentorNet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, 2018.

[17] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, 2020.

[18] Taehyeon Kim, Jongwoo Ko, JinHwan Choi, Se-Young Yun, et al. Fine samples for learning with noisy labels. In *Advances in Neural Information Processing Systems*, 2021.

[19] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[20] Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *International Conference on Learning Representations*, 2020.

[21] Junnan Li, Caiming Xiong, and Steven CH Hoi. Learning from noisy data with robust representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9485–9494, 2021.

[22] Junnan Li, Caiming Xiong, and Steven CH Hoi. Mopro: Webly supervised learning with momentum prototypes. In *International Conference on Learning Representations*, 2021.

[23] Shikun Li, Xiaobo Xia, Shiming Ge, and Tongliang Liu. Selective-supervised contrastive learning with noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 316–325, 2022.

[24] Wen Li, Limin Wang, Wei Li, Eirikur Agustsson, and Luc Van Gool. Webvision database: Visual learning and understanding from web data. *arXiv preprint arXiv:1708.02862*, 2017.

[25] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. In *Advances in Neural Information Processing Systems*, 2020.

[26] Xingjun Ma, Yisen Wang, Michael E Houle, Shuo Zhou, Sarah Erfani, Shutao Xia, Sudanthi Wijewickrema, and James Bailey. Dimensionality-driven learning with noisy labels. In *International Conference on Machine Learning*, pages 3355–3364. PMLR, 2018.

[27] Eran Malach and Shai Shalev-Shwartz. Decoupling "when to update" from "how to update". In *Advances in Neural Information Processing Systems*, 2017.

[28] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[29] Diego Ortego, Eric Arazo, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Multi-objective interpolation training for robustness to label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6606–6615, 2021.

[30] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[31] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

[32] Rasmus Rothe, Radu Timofte, and Luc Van Gool. Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision*, 126(2-4):144–157, 2018.

[33] Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

[34] Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. Joint optimization framework for learning with noisy labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5552–5560, 2018.

[35] Ryutaro Tanno, Ardavan Saeedi, Swami Sankaranarayanan, Daniel C Alexander, and Nathan Silberman. Learning from noisy labels by regularized estimation of annotator confusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11244–11253, 2019.

[36] Xinshao Wang, Yang Hua, Elyor Kodirov, and Neil M Robertson. IMAE for noise-robust learning: Mean absolute error does not treat examples equally and gradient magnitude's variance matters. *arXiv preprint arXiv:1903.12141*, 2019.

[37] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 322–330, 2019.

[38] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3733–3742, 2018.

[39] Xiaobo Xia, Tongliang Liu, Nannan Wang, Bo Han, Chen Gong, Gang Niu, and Masashi Sugiyama. Are anchor points really indispensable in label-noise learning? In *Advances in Neural Information Processing Systems*, 2019.

[40] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2691–2699, 2015.

[41] Yilun Xu, Peng Cao, Yuqing Kong, and Yizhou Wang. L_DMI: A novel information-theoretic loss function for training deep nets robust to label noise. In *Advances in Neural Information Processing Systems*, 2019.

[42] Yan Yan, Rómer Rosales, Glenn Fung, Ramanathan Subramanian, and Jennifer Dy. Learning from multiple annotators with varying expertise. *Machine Learning*, 95(3):291–327, 2014.

[43] Kun Yi and Jianxin Wu. Probabilistic end-to-end noise correction for learning with noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7017–7025, 2019.

[44] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In *International Conference on Machine Learning*, pages 7164–7173. PMLR, 2019.

[45] Xiyu Yu, Tongliang Liu, Mingming Gong, and Dacheng Tao. Learning with biased complementary labels. In *Proceedings of the European Conference on Computer Vision*, pages 68–83, 2018.

[46] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[47] Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in Neural Information Processing Systems*, 2018.

[48] Evgenii Zheltonozhskii, Chaim Baskin, Avi Mendelson, Alex M Bronstein, and Or Litany. Contrast to divide: Self-supervised pre-training for learning with noisy labels. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1657–1667, 2022.