Robust Dataset Condensation using Supervised Contrastive Learning

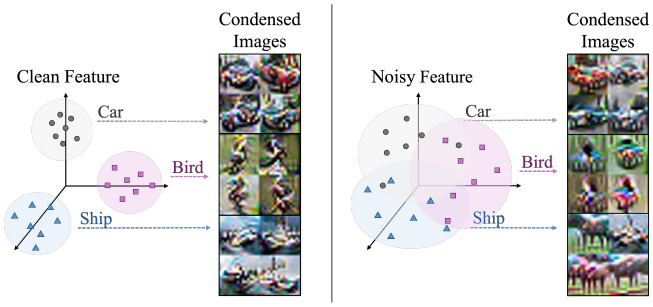
Nicole Hee-Yeon Kim Hwanjun Song*
Korea Advanced Institute of Science and Technology (KAIST)
Daejeon, Republic of Korea
{nicolekim, songhwanjun}@kaist.ac.kr

Abstract

Dataset condensation aims to compress large dataset into smaller synthetic set while preserving the essential representations needed for effective model training. However, existing methods show severe performance degradation when applied to noisy datasets. To address this, we present robust dataset condensation (RDC), an end-to-end method that mitigates noise to generate a clean and robust synthetic set, without requiring separate noise-reduction preprocessing steps. RDC refines the condensation process by integrating contrastive learning tailored for robust condensation, named golden MixUp contrast. It uses synthetic samples to sharpen class boundaries and to mitigate noisy representations, while its augmentation strategy compensates for the limited size of the synthetic set by identifying clean samples from noisy training data, enriching synthetic images with real-data diversity. We evaluate RDC against existing condensation methods and a conventional approach that first applies noise cleaning algorithms to the dataset before performing condensation. Extensive experiments show that RDC outperforms other approaches on CIFAR-10/100 across different types of noise, including asymmetric, symmetric, and real-world noise. Code is available at `https://github.com/DISL-Lab/RDC-ICCV2025`.

1. Introduction

The unprecedented growth of deep learning has raised significant concerns about computational resource sustainability in model training [45]. While various efficiency techniques like model compression [1] and data augmentation [17] have been proposed, their effectiveness remains limited. Dataset condensation emerges as a direct solution by distilling large-scale datasets into compact synthetic versions that preserve essential learning signals [43]. However, existing dataset condensation methods still have an unresolved critical issue: real-world datasets contain *noisy labels* that may not be true [40], significantly degrading the quality of condensed data. Consequently, existing condensation methods that assume clean, well-curated data face limitations in practical applications.



(a) Clean Dataset. (b) Noisy Dataset.

Figure 1. Condensed images generated by a dataset condensation method, Acc-DD [52], with CIFAR-10 datasets with clean and noisy labels, where the asymmetric label noise of 40% are injected following these rules: airplane→car, car→bird, and horse→ship.

Noise, commonly arising from data collection errors (e.g., sensor inaccuracies) [18], labeling errors (e.g., human or algorithmic misannotations) [25, 36], or external attacks (e.g., label flip attack) [47], directly impacts the effectiveness of dataset condensation, leading to significant performance degradation. Figure 1 illustrates the detrimental impact of noise on one of the state-of-the-art (SOTA) dataset condensation methods, Acc-DD [52]. In (a), the representations remain well-preserved in the absence of noise. However, in (b), with 40% of asymmetric noise, noise interference is evident. For instance, in the condensation process for the ship class, the presence of noise from the horse class results in a distorted representation, where the synthesized ship erroneously exhibits horse-like legs.

While two-stage approaches that apply dataset cleaning methods [4, 46] prior to condensation can help mitigate the problem, this approach is not a complete solution. Firstly, completely removing noise through dataset cleaning is inherently difficult, especially in large-scale datasets where subtle or systematic noise often persists [42]. Secondly, dataset cleaning processes are resource-intensive, requiring

*Corresponding author.

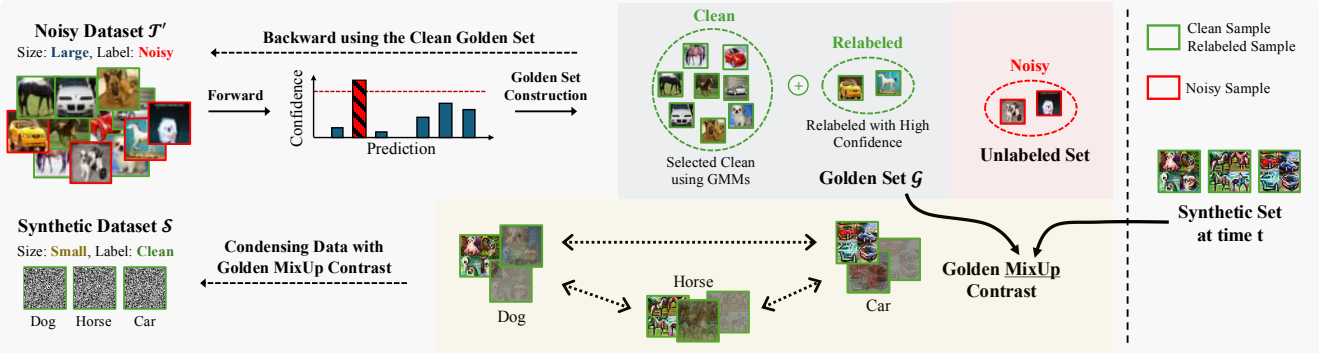


Figure 2. Robust dataset condensation using golden MixUp contrast. The noisy dataset is large but contains noisy labels, whereas the synthetic dataset is small but has clean, fixed labels. Our approach leverages the strengths of both contrasting aspects by mixing clean images selected from the noisy real dataset with the synthetic ones, enhancing diversity while minimizing the risk of label noise.

significant time and computational effort [9]. Thirdly, improper cleaning may inadvertently discard valuable information, further compromising the integrity of dataset [6]. Therefore, this underscores the need for an end-to-end dataset condensation method that is robust to label noise.

In this paper, we propose **Robust Dataset Condensation (RDC)**, the first dataset condensation method that maintains robust performance, even when applied to noisy datasets. In particular, unlike the two-stage approaches, RDC is implemented in an *end-to-end* manner, enabling the direct synthesis of a noise-resilient synthetic dataset without requiring any separate noise-handling procedure.

Specifically, RDC enhances the condensation process with a contrastive learning framework that leverages both synthetic and real data, as illustrated in Figure 2. Inspired by the fact that the condensed dataset is optimized with fixed labels, which are considered as *ground-truth* labels, RDC employs supervised contrastive learning [14] to directly learn the relationships among synthetic images in condensation. This facilitates class-wise feature separation, thereby minimizing interference from other classes and acting as an implicit regularization mechanism for the synthetic set. However, relying solely on synthetic data leads to a lack of diversity, particularly when the number of synthetic images per class (IPC) is low (e.g., IPC 10 or 50). This limitation prevents the condensation process from fully escaping the negative influence of label noise.

To remedy this, we introduce **golden MixUp contrast (GMC)**, a novel robust contrastive learning method tailored for dataset condensation, which addresses the lack of diversity caused by low IPC while minimizing the risk of label noise. Unlike naive contrastive learning, it systematically constructs a set of clean samples, called a *golden set*, which consists of samples selected or relabeled from the noisy training dataset, and integrates the golden set with synthetic samples with clean, fixed labels through MixUp. This approach not only enhances diversity but also ensures that noisy labels do not corrupt the contrastive learning process. By integrating GMC into the dataset condensation process, RDC allows the synthetic set to absorb correct contextual information from real samples, further improving its expressiveness and generalization ability.

Our main contributions can be summarized as follows:

- We are the first to address dataset condensation in the presence of label noise, a crucial advancement for real-world applications where noisy labels are inevitable.
- We introduce an end-to-end robust dataset condensation (RDC) framework that leverages the strengths of two contrasting datasets in dataset condensation, i.e., large, noisy real dataset versus small, yet clean synthetic dataset.
- We propose golden MixUp contrast to mitigate the issue of limited sample availability in the synthetic set while leveraging refined clean samples to enable the synthetic dataset to learn clean and diverse representations.
- Through extensive experiments, we demonstrate that our method outperforms existing dataset condensation approaches in noisy environments.

2. Related Work

Dataset Condensation. Dataset condensation [55] (also known as dataset distillation) aims to synthesize a small-scale dataset that retains the essential features of a large-scale original one, enabling models trained on the condensed data to achieve comparable accuracy to those trained on the full dataset. This technique has been widely explored in areas such as privacy-preserving machine learning [7, 19], continual learning [23, 24, 53–55], and federated learning [8, 41, 57]. Existing methods largely fall into four categories: meta-model matching, gradient matching, trajectory matching, and distribution matching [34]. Meta-model matching [43] optimizes a synthetic dataset by computing loss using the original dataset, with advancements such as kernel ridge regression [28] and Gram matrix-based optimization [58]. Gradient matching [55] aligns gradients between synthetic and real datasets, with improvements like Siamese augmentation (DSA) [53] and model perturbation

(Acc-DD) [52]. Trajectory matching [2] refines the synthetic dataset by following the training trajectory of models trained on the original dataset, with enhancements including soft label assignment (TESLA) [5] and trajectory range adjustment (DATM) [11]. Distribution matching [54] aligns the distributions of synthetic and original datasets, further improved through class-aware regularization (IDM) [56] and attention map matching (DataDAM) [35].

While these methods demonstrate strong performance on clean datasets, their effectiveness significantly degrades when applied to noisy datasets. Therefore, to address this issue, we propose a condensation method that remains robust even in the presence of noise.

Contrastive Learning. Contrastive learning is a self-supervised technique that forms positive and negative pairs in feature space, bringing positive pairs closer while pushing negative pairs apart. Methods such as SimCLR [3] and MoCo [13] use this principle to learn discriminative features. Supervised contrastive learning (SupCon) [14] extends this approach by utilizing label information, treating all samples of the same class as positive pairs and those of different classes as negative pairs. SupCon has been applied in LLM training [10, 37], multimodal representation learning [26, 27], robustness to noisy labels [21, 30], and recommendation systems [48]. MixUp contrast (MixCo) [15] is another extension that integrates MixUp [50] into contrastive learning, enabling the use of not only positive pairs but also semi-positive pairs to more effectively distinguish the positive pairs from negative pairs. The key idea behind MixCo is to incorporate softened data representations into contrastive learning, allowing the model to capture implicit relationships between positive and negative samples while alleviating the instance discrimination problem.

But when MixCo is applied to noisy data, it fails to filter out noise and instead enforces contrastive learning on incorrect labels. To address this, we propose a novel approach that enables contrastive learning on reliable samples.

3. Preliminaries

Formulation of Dataset Condensation. Suppose $\mathcal{T} = \{(x_1, y_1), \ldots, (x_{|\mathcal{T}|}, y_{|\mathcal{T}|})\}$ denotes the original dataset, where x_i is an image and y_i is its corresponding label. Similarly, let $\mathcal{S} = \{(s_1, y_1^s), \ldots, (s_{|\mathcal{S}|}, y_{|\mathcal{S}|}^s)\}$ be the significantly smaller synthetic dataset, where $|\mathcal{T}| \gg |\mathcal{S}|$ and $|\cdot|$ denotes the dataset size. Then, the objective of dataset condensation is to ensure that a model trained on $\mathcal{S}$ achieves performance comparable to one trained on $\mathcal{T}$.

Dataset condensation methods relies on a model trained on $\mathcal{T}$ to extract representations that serve as the foundation for condensation. Given a model f_θ, the parameters are optimized to minimize the cross-entropy (CE) loss as:

$$\theta^* = \arg\min_\theta \frac{1}{|\mathcal{T}|} \sum_{(x,y)\in\mathcal{T}} \ell_{\text{CE}}\big(f_\theta(x), y\big). \tag{1}$$

This optimization seeks to find θ^* that minimizes the loss, ensuring that the model learns the best possible representation from the dataset $\mathcal{T}$.

Once the model has been trained on $\mathcal{T}$, dataset condensation methods construct a synthetic dataset $\mathcal{S}$ such that a model trained on $\mathcal{S}$ learns a representation similar to that obtained from $\mathcal{T}$. To achieve this, a condensation loss $\mathcal{L}_{\text{cond}}$ is introduced, which aligns the representations learned from the model trained on $\mathcal{S}$ with those from the reference model trained on $\mathcal{T}$. The specific formulation of $\mathcal{L}_{\text{cond}}$ varies depending on the matching strategy employed. To define the training process on $\mathcal{S}$, an auxiliary model is optionally required and trained on the learnable synthetic dataset $\mathcal{S}$ as:

$$\theta_{\mathcal{S}}^* = \arg\min_{\theta_{\mathcal{S}}} \frac{1}{|\mathcal{S}|} \sum_{(s,y^s)\in\mathcal{S}} \ell_{\text{CE}}\big(f_{\theta_{\mathcal{S}}}(s), y^s\big). \tag{2}$$

Then, the synthetic dataset $\mathcal{S}$ is then optimized by minimizing the condensation loss:

$$\mathcal{S}^* = \arg\min_{\mathcal{S}} \mathcal{L}_{\text{cond}}(f_{\theta^*}(\mathcal{T}), f_{\theta_{\mathcal{S}}^*}(\mathcal{S})), \tag{3}$$

where $f_{\theta^*}(\mathcal{T})$ and $f_{\theta_{\mathcal{S}}^*}(\mathcal{S})$ denote the outputs of the function f applied to $\mathcal{T}$ and $\mathcal{S}$, respectively. The extracted features are obtained by utilizing specific components of these outputs, based on the matching method. Some approaches like IDM [56] use a shared model for both $\mathcal{T}$ and $\mathcal{S}$.

The formulation above assumes that the original dataset $\mathcal{T}$ is clean. Yet, real-world data often contains label noise, as $\mathcal{T}' = \{(x_1, \tilde{y}_1), \ldots, (x_{|\mathcal{T}'|}, \tilde{y}_{|\mathcal{T}'|})\}$, where part of $\tilde{y}$ are incorrect labels, while the rest remain correct. When clean label y is replaced with its noisy counterpart $\tilde{y}$, the optimization steps involving θ^* through Eq. (1) results in overfitting to the noise. Consequently, the representations extracted from $f_{\theta^*}(\mathcal{T}')$ become degraded, which significantly deteriorates the quality of $\mathcal{S}^*$ obtained in Eq. (3).

To address this issue, it is necessary to ensure that Eqs. (1) and (3) remain robust to label noise in the presence of $\mathcal{T}'$. Specifically, the goal is to train $\mathcal{S}$ in a manner that mitigates the impact of noisy labels $\tilde{y}$, enabling it to generalize as if $\mathcal{T}'$ contained only true labels. This ensures that the synthetic dataset $\mathcal{S}$ achieves comparable performance to that obtained when trained on a clean dataset.

Formulation of Supervised Contrastive Learning. For the original dataset, $\mathcal{T} = \{(x_1, y_1), \ldots, (x_{|\mathcal{T}|}, y_{|\mathcal{T}|})\}$, supervised contrastive learning (SupCon) establishes an anchor (x_i, y_i) as the central reference point. It assigns samples with the same label as the anchor to positive pairs, where positive set is defined as $P = \{(x_p, y_p) \in \mathcal{T} \mid$

$x_p \neq x_i,\ y_p = y_i\}$. Meanwhile, samples with different labels are designated as negative pairs, where the negative set is given by $N = \{(x_n, y_n) \in \mathcal{T} \mid y_n \neq y_i\}$. By optimizing the model to bring the anchor closer to positive pairs and push it further away from negative pairs, the class boundaries can be reinforced. Since SupCon is performed in the embedding space, embeddings are extracted for all samples using the model f_θ. Utilizing these embeddings, the overall loss function $\mathcal{L}_{\text{SupCon}}$ is obtained by summing the individual losses computed for each sample as an anchor as:

$$\mathcal{L}_{\text{SupCon}}(\mathcal{T}, P, N) = \sum_{x_i \in \mathcal{T}} \frac{-1}{|P|} \sum_{x_p \in P} \log \frac{\exp(f_\theta(x_i) \cdot f_\theta(x_p)/\tau)}{\sum_{x_a \in P \cup N} \exp(f_\theta(x_i) \cdot f_\theta(x_a)/\tau)}, \quad (4)$$

where τ is a temperature parameter that controls the sharpness of the similarity distribution by scaling the logits.

4. Robust Dataset Condensation (RDC)

Overview. We regularize the dataset condensation process by incorporating supervised contrastive learning (SupCon), leveraging the fact that *the fixed labels assigned to synthetic data in the condensation process are inherently regarded as ground truths*. This enables effective representation learning, ensuring that synthetic samples maintain well-structured class boundaries while mitigating the impact of noisy labels in the real dataset. However, a major bottleneck of this approach is the severe *scarcity* of synthetic samples due to the inherent constraints of dataset condensation. Thus, we aim to address this limitation by enhancing the diversity of the synthetic set.

To this end, we conduct supervised contrastive learning based on the ground truth labels assigned to the synthetic samples while integrating carefully selected clean samples from the original noisy dataset to enhance diversity and mitigate the risk of label noise. The following section details the key components of our RDC method.

4.1. Golden MixUp Contrast (GMC)

To mitigate feature distortion from noisy labels in the original dataset, we employ SupCon on *synthetic* set. Since the labels y_i^s in the synthetic dataset $\mathcal{S} = \{(s_1, y_1^s), \ldots, (s_{|\mathcal{S}|}, y_{|\mathcal{S}|}^s)\}$ are always true and fixed, we leverage this label information as the supervisory signal to apply SupCon. However, the effective application of SupCon requires a sufficient number of samples. Since condensation generates only a limited number of images per class, effective augmentation is crucial to compensate for the lack of diversity in the context.

To address this, we propose *golden MixUp contrast*, which complements the limited diversity of synthetic samples by mixing the real samples selected from the original training dataset. Note that naively selecting real images for MixUp may introduce noise, as mixed samples can contain conflicting or misleading semantic information. Therefore, we construct a *golden set* $\mathcal{G}$, consisting of carefully selected clean samples $\mathcal{C}$ and re-labeled confident samples $\mathcal{R}$ with high confidence. This enables reliable MixUp, achieving high diversity while minimizing the negative impact of incorrect labels.

4.1.1. Extracting Golden Set from Noisy Dataset

During the training of the model f_θ in Eq. (2), after initial warm-up epochs t_0, we accumulate the loss values of each training sample until the ongoing epoch T using a model trained on all samples from the noisy original dataset $\mathcal{T}'$. The accumulated loss list L is defined as $L = \{\ell_i \mid i \in I'\}$, where I' represents the set of sample indices, specifically, $I' = \{1, \ldots, |\mathcal{T}'|\}$. The accumulated loss ℓ_i for sample x_i is given by $\ell_i = \sum_{t=t_0}^{T} \ell_{\text{CE}}(f_{\theta_t}(x_i), \tilde{y}_i)$.

After applying a logarithmic transformation to the accumulated loss values, the accumulated loss list L turns to L_{log}, defined as $L_{log} = \{\log \ell_i \mid i \in I'\}$. Subsequently, we fit a bi-modal univariate Gaussian Mixture Model (GMM) with two components, where the lower-mean distribution is classified as the *clean* set, and the higher-mean distribution is classified as the *unclean* set. This is based on the observation that clean labels tend to have lower loss values than noisy ones, attributed to the *memorization effect* of deep neural networks [20, 39, 40, 49].

Formally, the log-transformed accumulated loss values are modeled as a mixture of two Gaussian distributions:

$$p(\log \ell_i) = \pi_1 \mathcal{N}(\log \ell_i \mid \mu_1, \sigma_1^2) + \pi_2 \mathcal{N}(\log \ell_i \mid \mu_2, \sigma_2^2), \quad (5)$$

where $\mathcal{N}(\log \ell_i \mid \mu_k, \sigma_k^2)$ is a Gaussian distribution with mean μ_k and variance σ_k^2, and π_1, π_2 are the mixing coefficients such that $\pi_1 + \pi_2 = 1$. We assume $\mu_1 < \mu_2$, meaning the lower-mean Gaussian corresponds to the clean set, and the higher-mean Gaussian corresponds to the unclean set. Each sample x_i is assigned to one of the two components based on its posterior probability:

$$p(z_i = k \mid \log \ell_i) = \frac{\pi_k \mathcal{N}(\log \ell_i \mid \mu_k, \sigma_k^2)}{\sum_{j=1}^{2} \pi_j \mathcal{N}(\log \ell_i \mid \mu_j, \sigma_j^2)}, \quad (6)$$

where $z_i \in \{1, 2\}$ represents the component assignment. The clean set $\mathcal{C}$ and unclean set $\mathcal{U}$ are then defined as:

$$\begin{aligned} \mathcal{C} &= \{(x_i, \tilde{y}_i) \in \mathcal{T}' \mid p(z_i = 1 \mid \log \ell_i) > 0.5\} \\ \mathcal{U} &= \{(x_i, \tilde{y}_i) \in \mathcal{T}' \mid p(z_i = 2 \mid \log \ell_i) \geq 0.5\}. \end{aligned} \quad (7)$$

$\mathcal{C}$ consists of samples more likely to belong to the lower-mean Gaussian (clean set), and $\mathcal{U}$ consists of samples more likely to belong to the higher-mean Gaussian (unclean set).

Additionally, before overfitting to noisy labels, a model's predictions can provide valuable insights into the true underlying labels [31, 39, 51]. Thus, for all samples in $\mathcal{U}$,

we pass them through the model f_θ and obtain their softmax values across all classes. If a sample exhibits a significantly high softmax value for a single class beyond a certain threshold, we consider it to have high confidence. These samples undergo relabeling, as the model is able to correctly identify easy samples that were originally misclassified due to label noise. After relabeling, these samples have fixed labels, y_i^{fix} which are assumed to be correct, and are collectively referred to as the *relabeled* set $\mathcal{R}$:

$$\mathcal{R} = \left\{ (x_i, y_i^{\text{fix}}) \mid (x_i, \tilde{y}_i) \in \mathcal{U} \wedge \max p_\theta(y \mid x_i) > c \right\}, \quad (8)$$

$$\text{where } y_i^{\text{fix}} = \arg\max_y p_\theta(y \mid x_i). \quad (9)$$

Here, c is set to be 0.95 (i.e., very high confidence) following the recent methods in noise-robust learning [38, 39].

Finally, the *golden* set $\mathcal{G}$ is defined as the union of the clean set $\mathcal{C}$ and the relabeled set $\mathcal{R}$. Since $\mathcal{G}$ is assumed to contain only correct labels, we attach g to the label as:

$$\mathcal{G} = \mathcal{C} \cup \mathcal{R} = \left\{ (x_1, y_1^g), \ldots, (x_{|\mathcal{G}|}, y_{|\mathcal{G}|}^g) \right\}. \quad (10)$$

We provide a detailed analysis on the golden set in terms of its size and correctness in Appendix A.

4.1.2. SupCon with Golden Set using MixUp

By utilizing the golden set defined in Eq. (10), we perform MixUp between synthetic samples (s_i, y_i^s) and samples from the golden set (x_j, y_j^g), which is formulated as:

$$s_i^* = \lambda s_i + (1 - \lambda) x_j, \quad (\lambda \in [0, 1]), \quad (11)$$

where λ is the MixUp coefficient controlling the interpolation between the two samples. The set of all mixed samples s_i^* is denoted as $\mathcal{S}_{\text{MixUp}}$. In golden Mixup contrast, the anchor is set as the mixed sample, s_i^*. For each anchor sample, positive and negative pairs for SupCon are defined with two criteria, as the label of s_i^* is affected by the two samples involved in MixUp: (1) a sample s_i from the synthetic set $\mathcal{S}$ with the label y_i^s, and (2) a sample x_j from the golden set $\mathcal{G}$ with the label y_j^g. The bidirectional alignment in SupCon using the two criteria preserves consistency between mixed–synthetic and mixed–original sample pairs, preventing feature distortions and enhancing robustness [15].

Pairs with Synthetic Set. When defining positive pairs based on the synthetic set, they can be formulated as $P_s^* = \{(s_p, y_p^s) \in \mathcal{S} \mid y_p^s = y_i^s\}$, which corresponds to samples in $\mathcal{S}$ that share the same label y_i^s as the synthetic sample s_i associated with the anchor s_i^*. Negative pairs refer to samples in the synthetic set that do not share the same label as s_i and are defined as $N_s^* = \{(s_n, y_n^s) \in \mathcal{S} \mid y_n^s \neq y_i^s\}$. Based on this definition, SupCon loss with respect to the synthetic set is formulated as follows:

$$\mathcal{L}_{\text{contrast}}^{\text{syn}} = \mathcal{L}_{\text{SupCon}}(\mathcal{S}_{\text{MixUp}}, P_s^*, N_s^*). \quad (12)$$

Pairs with Golden Set. When defining positive pairs with the golden set, they are expressed as $P_g = \{(x_p, y_p^g) \in \mathcal{G} \mid y_p^g = y_j^g\}$, which represents samples in $\mathcal{G}$ that share the same label y_j^g as the golden sample x_j associated with the anchor s_i^*. Negative pairs consist of samples in the golden set that have a different label from x_j and are defined as $N_g^* = \{(x_n, y_n^g) \in \mathcal{G} \mid y_n^g \neq y_j^g\}$. With this definition, the SupCon loss for the golden set is formulated as:

$$\mathcal{L}_{\text{contrast}}^{\text{golden}} = \mathcal{L}_{\text{SupCon}}(\mathcal{S}_{\text{MixUp}}, P_g^*, N_g^*). \quad (13)$$

Thus, the final loss for golden MixUp contrast is defined as follows. The weights used in MixUp are applied to $\mathcal{L}_{\text{contrast}}^{\text{syn}}$ and $\mathcal{L}_{\text{contrast}}^{\text{golden}}$, and their weighted sum forms the final loss. Additionally, we incorporate $\mathcal{L}_{\text{syn}} = \mathcal{L}_{\text{SupCon}}(\mathcal{S}, P, N)$, which is computed using only the synthetic set, $\mathcal{S}$[1]. This approach strengthens the compactness of synthetic images by applying contrastive learning solely within the synthetic set. Formally, the GMC loss is:

$$\mathcal{L}_{\text{GMC}} = (\lambda \cdot \mathcal{L}_{\text{contrast}}^{\text{syn}} + (1 - \lambda) \cdot \mathcal{L}_{\text{contrast}}^{\text{golden}}) + \mathcal{L}_{\text{syn}}. \quad (14)$$

This enables the synthetic set to incorporate the diverse contexts of real images with correct labels, resulting in a clear distinction between class representation boundaries in dataset condensation, even in the presence of label noise.

4.2. Improving Robustness of Model Training

Furthermore, we enhance the robustness of the training model f_θ in Eq. (1) while improving the condensation process in Eq. (2) by our golden MixUp contrast. During the training of f_θ, we employ semi-supervised learning using the constructed golden set $\mathcal{G}$. Specifically, since the golden set $\mathcal{G}$ is clean, it can be defined as the labeled set. The remaining unclean samples, excluding the relabeled ones, i.e., $\mathcal{U}/\mathcal{R}$, are considered noisy and are therefore defined as the unlabeled set, where label information is not utilized. By leveraging this unlabeled set, the model is enabled to extract clean features even from noisy datasets, further enhancing the quality of the synthetic set.

To achieve this, we design a loss function for the model f_θ to perform semi-supervised learning. The loss function consists of that used in DivideMix [33] (see Appendix B.1.1 for details), along with the cross-entropy loss of the golden set, $\mathcal{L}_{\mathcal{G}} = \ell_{\text{CE}}(\mathcal{G})$. Therefore, the loss related to model updates in Eq. (1) is updated as follows:

$$\mathcal{L}_{\text{model}}^{\text{RDC}} = \mathcal{L}_{\text{DivideMix}} + \mathcal{L}_{\mathcal{G}}. \quad (15)$$

4.3. Objective Function of RDC

To summarize, we modify the training of the model in Eq. (1) and the learning of condensed images in Eq. (2), both of which are vulnerable to label noise.

[1] To increase the number of images, augmentation such as controlling color, crop, cutout, flip, scale, and rotate is applied to the synthetic dataset.

Firstly, the noisy-resilient training of the model through semi-supervised learning is formulated as follows:

$$\theta_{\text{RDC}} = \arg\min_{\theta}(\mathcal{L}^{\text{RDC}}_{\text{model}}). \tag{16}$$

Through this optimization, the model mitigates overfitting caused by noise and extract accurate representations.

Secondly, the noise-resilient learning of the condensed images is denoted as follows:

$$\mathcal{S}_{\text{RDC}} = \arg\min_{\mathcal{S}}(\mathcal{L}_{\text{cond}} + \mathcal{L}_{\text{GMC}}), \tag{17}$$

where the loss from the golden MixUp contrast is combined with the loss defined by the chosen dataset condensation method. Note that our RDC method is flexible and can be plugged into various condensation approaches. The algorithm of RDC is provided in Appendix C.

5. Experiment

Following the evaluation protocol from previous dataset condensation studies [53, 54, 56], we evaluate the performance of RDC on an image classification task. The evaluation pipeline involves learning the condensed images and training classifiers from scratch on them. Note that, in our setup, training datasets for dataset condensation are corrupted by either artificial or real-world label noise. The final test accuracy is a commonly used measure of robustness to noisy labels [20, 39]. In other words, higher test accuracy indicates greater robustness. To ensure reliability, we repeat each test three times and report the average test accuracy.

Additionally, we present the evaluation results using Dd-Ranking [22] in Appendix D.

Noisy Datasets. We conduct experiments using both the original clean datasets, CIFAR-10 and CIFAR-100 [16], as well as their noisy counterparts. Specifically, we consider three types of label noise: (1) *asymmetric* noise [32], where labels are flipped to its adjacent class (e.g., class 0 $\rightarrow$ class 1, class 1 $\rightarrow$ class 2); (2) *symmetric* noise [32], where labels are randomly flipped to any other class with equal probability; and (3) CIFAR-10N and CIFAR-100N with *real-world* noise [44], where their labels are annotated by humans and contains mistakes due to subjective judgement or ambiguity. For the two former synthetic noise, we adjust the noise ratio from 0% (clean) to 20% (moderate noise) and 40% (heavy noise) to evaluate the impact of different noise levels on dataset condensation. For the latter real-world noise, we select CIFAR-10N-{Random1, Worse} with 17% and 40% of noise, and CIFAR-100N-Noisy with 40% noise.

Dataset Condensation Setups. RDC is compatible with existing dataset condensation methods. Thus, we implement RDC on top of two popular condensation methods, each taking a different approach: IDM [56] (see Appendix B.3), the SOTA method based on distribution matching; and Acc-DD [52] (see Appendix B.4), another method based on gradient matching. In our experiment, IDM is used mainly because of its simplicity compared to other condensation methods. Specifically, IDM shares a single network for both the original and synthetic datasets in dataset condensation, and thus the extraction of the golden set and its subsequent use in semi-supervised learning directly influence the training of the synthetic dataset. These characteristics make IDM particularly well-suited to maximizing the advantages of RDC. The results with Acc-DD can be found in Section 5.2. For the dataset condensation setup, we learn 10 and 50 condensed images per class (Img/Cls) with noisy CIFAR-10, while we learn 1 and 10 Img/Cls with noisy CIFAR-100.

Regarding the backbone, we mainly use ResNet18 [12], a model popularly used in noise-robust learning, throughout the experiment section. To verify the generalization capability of RDC, we present results using alternative backbones, 3-layer ConvNet and VGG11, in Section 5.4.

Compared Methods. Since our method is built on top of two existing methods, we directly compare it with a random selection approach (denoted as Random) and the two base methods, IDM and Acc-DD, as well as their extension using the two-stage approach[2]. Specifically, for the two-stage approach (see Appendix B.2), we first apply DivideMix [20], a state-of-the-art semi-supervised robust learning method, before dataset condensation to refine the noisy training set.

Implementation Details. We primarily follow the implementation of our base methods, IDM and Acc-DD. Regarding the hyperparameters introduced by RDC, we set the warm-up epochs to be 5 and 20 for noisy CIFAR-10 and CIFAR-100 datasets, respectively. For the golden MixUp contrast, we set the MixUp λ value to 0.75 (found by a grid search). For the SupCon loss in Eq. (4), the temperature τ is set to be 0.07 in every case. For the DivideMix method, which is used in the two-stage extension and in Eq. (15), we follow the original settings summarized in Table 7 of [20]. Further details on grid search and implementation can be found in Appendix B.1 of the supplementary material.

5.1. Robustness against Label Noise

Tables 1 and 2 present a robustness comparison of our method against three baseline methods, along with reference results (Whole Dataset) serving as an upper bound, on noisy CIFARs datasets. We include the random selection (Random), a coreset-based approach that selects random images from the original set, following prior studies. RDC achieves a significant performance improvement on both datasets. Notably, the improvement is even more

[2]GMC can also be applied to soft-label dataset condensation methods, DATM [11], by using similarity scores weighted by class probability products. More detailed formulations are provided in Appendix E.

CIFAR-10	Clean		Asymmetric Noise				Symmetric Noise				Real-world Noise			
Noise Ratio	≈ 0%		20%		40%		20%		40%		Random1 (17%)		Worse (40%)	
Img/Cls	10	50	10	50	10	50	10	50	10	50	10	50	10	50
Random	22.28	36.05	19.55	29.12	19.53	24.48	17.11	29.31	18.36	24.28	20.19	31.88	20.19	28.26
IDM	45.10	60.24	39.63	47.64	30.61	34.85	42.97	56.54	41.17	45.62	44.68	57.19	38.36	49.46
IDM + Two-stage	43.77	60.24	45.59	60.23	33.29	38.61	45.70	59.27	**46.52**	59.26	45.25	60.15	45.29	60.10
IDM + RDC (Ours)	**47.28**	**60.80**	**46.92**	**61.76**	**41.35**	**55.55**	**47.12**	**63.15**	46.23	**59.61**	**48.36**	**61.85**	**46.52**	**61.93**
Whole Dataset	95.37		81.42		58.81		84.00		64.79		85.45		67.36	

Table 1. Robustness comparison between random selection and three dataset condensation methods on **noisy CIFAR-10** datasets under various noise conditions. All methods use ResNet18 as the backbone. Img/Cls denotes the number of images per class and Whole Dataset denotes the model accuracy on the entire training set, which is the upper bound for dataset condensation with varying noise types.

CIFAR-100	Clean		Asymmetric Noise				Symmetric Noise				Real-world Noise	
Noise Ratio	≈ 0%		20%		40%		20%		40%		Noisy (40%)	
Img/Cls	1	10	1	10	1	10	1	10	1	10	1	10
Random	3.73	12.46	3.04	11.20	2.79	6.83	2.94	10.40	2.46	7.14	3.21	10.27
IDM	5.16	18.94	4.23	18.04	2.57	12.42	3.97	19.02	2.69	16.59	4.09	16.35
IDM + Two-stage	5.81	18.97	5.59	19.75	3.44	15.57	6.08	19.51	5.77	**19.85**	6.03	19.28
IDM + RDC (Ours)	**11.23**	**26.73**	**8.83**	**23.42**	**6.35**	**15.60**	**9.64**	**24.04**	**6.00**	17.60	**9.60**	**24.62**
Whole Dataset	78.22		65.98		46.37		64.67		49.45		54.41	

Table 2. Robustness comparison between random selection and three dataset condensation methods on **noisy CIFAR-100** datasets under various noise conditions. All methods use ResNet18 as the backbone.

Tiny-ImageNet	Clean		Asymmetric 40%		Symmetric 40%	
Img/Cls	1	10	1	10	1	10
Random	1.93	7.90	1.20	3.90	1.39	3.97
IDM	2.32	7.07	2.08	6.38	2.26	6.61
IDM + Two-stage	0.61	0.54	0.67	0.51	0.60	0.59
IDM + RDC (Ours)	**3.41**	**11.02**	**3.65**	**11.09**	**3.85**	**11.73**
Whole Dataset	65.58		40.06		36.66	

Table 3. Robustness comparison between random selection and three dataset condensation methods on noisy Tiny-ImageNet under various noise types. All methods use ResNet18 as backbone.

pronounced on CIFAR-100, where dataset condensation is more challenging due to the larger number of classes.

Specifically, RDC exhibits a significant improvement, surpassing the two-stage approach across almost all IPC settings and noise conditions. Particularly, in the CIFAR-10 setting with 40% asymmetric noise, where the two-stage approach fails to fully recover performance, our method achieves an accuracy of 55.55%, bringing it much closer to the 60.24% accuracy obtained by IDM on the clean dataset (see the Clean column). This reveals that the stability of our approach and highlights its generalizability in challenging asymmetric noisy setup. Moreover, our method not only restores performance in noisy environments but also surpasses other methods trained on clean datasets. This is likely attributed to the presence of label noise even in clean CIFAR datasets [29, 44] and the regularization effect introduced by applying SupCon within the dataset condensation process.

In contrast, IDM suffers a substantial performance drop when compared to its performance on clean datasets, due to its lack of consideration for noise. The two-stage approach is effective in recovering performance in most noisy environments. However, it fails to fully restore performance when 40% asymmetric noise was present in both CIFAR-10 and CIFAR-100, indicating that the insufficiency of performing data cleaning in a separate stage without integrating it into an end-to-end process.

Furthermore, RDC remains its robust performance even on a larger noisy Tiny-ImageNet dataset in Table 3.

5.2. Implementation with Acc-DD

In Table 4, we apply RDC to Acc-DD, the condensation method in gradient matching. We synthesize 10 Img/Cls on noisy CIFAR-10 using ResNet18. The results remain consistent when using IDM as the base model. In detail, Acc-DD is highly vulnerable to noisy settings. While the two-stage approach aids performance recovery, it is insufficient when the dataset contained 40% asymmetric noise. However, when RDC is integrated into Acc-DD, the performance (60.99%) exceeds that of Acc-DD trained on a clean dataset (57.18%) in all noise settings except the 40% asymmetric noise case. These results highlight the applicability of RDC across different dataset condensation approaches. Additional experiments on applying RDC to MTT [2] and DATM [11] are presented in Appendix F.

5.3. Component Ablation Study

Table 5 presents the stepwise component analysis of RDC on noisy CIFAR-10 with varying noise types with 10 Img/Cls. Using IDM as the base model, we progressively activate each component of RDC in the order of (1) applying semi-supervised learning (DivideMix) in Eq. (15), (2)

CIFAR-10	Clean	Asymmetric Noise		Symmetric Noise		Real-world Noise	
Noise Ratio	$\approx 0\%$	20%	40%	20%	40%	Ran1 (17%)	Worse (40%)
Random	22.28	19.55	19.53	17.11	18.36	20.19	20.19
Acc-DD	57.18	49.42	35.33	50.77	40.40	53.79	47.27
Acc-DD + Two-stage	57.93	58.93	40.97	58.85	57.72	58.34	58.31
Acc-DD + RDC (Ours)	**60.99**	**59.47**	**55.18**	**60.12**	**58.62**	**58.84**	**59.39**
Whole Dataset	95.37	81.42	58.81	84.00	64.79	85.45	67.36

Table 4. Robustness comparison among different methods using **Acc-DD as the base model** on CIFAR-10 under various noise conditions.

Component	Asymmetric	Symmetric	Real-world
IDM	30.61	41.17	38.36
+ (1) SSL (DivideMix)	35.23	41.93	44.47
+ (2) SupCon wo. Augment	33.89	43.63	44.99
+ (3) SupCon w. Augment	35.24	44.94	45.58
+ (4) Golden MixUp Contrast	41.35	46.23	46.52

Table 5. Ablation study on noisy CIFAR-10 using ResNet18. The last row (all components applied) corresponds to RDC.

Backbone	VGG11		ResNet18	
Eval.\Cond.	IDM	IDM + RDC	IDM	IDM + RDC
ConvNet	26.83	38.39	38.25	57.46
VGG11	28.56	36.59	37.02	55.57
ResNet18	24.61	38.96	34.85	57.77

Table 6. Generalization of RDC to various model architectures.

incorporating only the canonical SupCon $\mathcal{L}_{\text{syn}}$ in Eq. (14), (3) utilizing simple augmentation such as flip and crop on the synthetic set, and (4) applying golden MixUp contrast.

The results reveal that: (1) the use of DivideMix using the golden set $\mathcal{G}$ results in a performance improvement. Subsequently, (2) the use of SupCon using only the synthetic set $\mathcal{S}$ leads to an improvement; however, the performance is deteriorated (in Asymmetric) or the gain is insignificant (in Symmetric and Real-world). (3) Simple data augmentation fails to address the diversity deficiency in the synthetic set for SupCon, as indicated by its minimal impact on performance, while (4) the application of GMC considerably boosts the performance. Thus, the ablation underscores the synergistic impact of all RDC components.

5.4. Generalization to Other Architectures

Table 6 presents the generalizability of RDC across various model architectures. Specifically, we synthesize 50 Img/Cls using two different backbones, VGG11 and ResNet18, on CIFAR-10 with 40% asymmetric noise. We then report the classification accuracy of three models, including ConvNet, VGG11, and ResNet18, each trained using the condensed images generated by either VGG11 or ResNet18. Firstly, the results reveal that our RDC method maintains its robustness when condensing images with VGG11 other than ResNet18 (as evidenced by the improvement in the IDM and IDM+RDC sub-columns under the VGG11 column). Secondly, the condensed images, regardless of whether they are generated from VGG11 or ResNet18, maintain their effectiveness in cross-architecture setups, e.g., condensed images from VGG11 can be used to train ConvNet or ResNet18 (as evidenced by each column). Here, note that the improvement achieved by RDC persists even in the cross-architecture setup. Therefore, RDC generalizes well across different model architectures.

(a) Acc-DD. (b) w. Two-stage. (c) w. RDC.

Figure 3. Visualization of condensed images, comparing Acc-DD and its extension using the two-stage and RDC (ours) methods. Appendix H provides more examples.

5.5. Comparison with Condensed Images

Figure 3 visualizes a subset of 10 Img/Cls condensed images when CIFAR-10 contains 40% asymmetric noise. In (a), where Acc-DD is directly applied for condensation, interference from other classes is evident, i.e., frog-like features appear in the condensed images for horse (top row), and horse-like features are present in the condensed images for ship (bottom row). In (b), a two-stage approach is employed, where dataset cleaning is performed before Acc-DD. While this mitigates some interference compared to (a), imperfect dataset cleaning still results in residual contamination from other class representations. In contrast, (c) demonstrates that when RDC is integrated into Acc-DD, interference from other classes is clearly eliminated, enabling the accurate synthesis of horse and ship representations.

6. Conclusion

We propose RDC, the first solution to address the negative impact of noisy labels on dataset condensation. Noisy training data blurs class boundaries, resulting in representations affected by interference from other classes. To mitigate this, we use the fixed labels of the synthetic set to apply supervised contrastive learning, while introducing golden MixUp contrast, which transfers the representations of reliable real images to the synthetic set. Through experiments, we show the robustness and generalizability of RDC across various noise types, levels, and architectures.

Acknowledgements. This work was supported by the NRF grant funded by the Korea government (MSIT) (No. RS-2024-00334343, A System for Enhancing Language Model Reliability with High-Quality Data and Automated Quality Assessment) and the IITP grant funded by the Korea government (MSIT) (No. RS-2024-00445087, Enhancing AI Model Reliability Through Domain-Specific Automated Value Alignment Assessment).

References

[1] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *SIGKDD*, 2006.

[2] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A Efros, and Jun-Yan Zhu. Dataset distillation by matching training trajectories. In *CVPR*, 2022.

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[4] Xu Chu, Ihab F Ilyas, Sanjay Krishnan, and Jiannan Wang. Data cleaning: Overview and emerging challenges. In *SIGMOD*, 2016.

[5] Justin Cui, Ruochen Wang, Si Si, and Cho-Jui Hsieh. Scaling up dataset distillation to imagenet-1k with constant memory. In *ICML*, 2023.

[6] Tamraparni Dasu and Ji Meng Loh. Statistical distortion: Consequences of data cleaning. *arXiv preprint arXiv:1208.1932*, 2012.

[7] Tian Dong, Bo Zhao, and Lingjuan Lyu. Privacy for free: How does dataset condensation help privacy? In *ICML*, 2022.

[8] Jack Goetz and Ambuj Tewari. Federated learning via synthetic data. *arXiv preprint arXiv:2008.04489*, 2020.

[9] Kartikay Goyle, Quin Xie, and Vakul Goyle. Dataassist: A machine learning approach to data cleaning and preparation. In *IntelliSys*, 2024.

[10] Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. Supervised contrastive learning for pre-trained language model fine-tuning. *arXiv preprint arXiv:2011.01403*, 2020.

[11] Ziyao Guo, Kai Wang, George Cazenavette, Hui Li, Kaipeng Zhang, and Yang You. Towards lossless dataset distillation via difficulty-aligned trajectory matching. In *ICLR*, 2024.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

[14] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *NeurIPS*, 2020.

[15] Sungnyun Kim, Gihun Lee, Sangmin Bae, and Se-Young Yun. Mixco: Mix-up contrastive learning for visual representation. *arXiv preprint arXiv:2010.06300*, 2020.

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

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.

[18] Daoliang Li, Ying Wang, Jinxing Wang, Cong Wang, and Yanqing Duan. Recent advances in sensor fault diagnosis: A review. *Sensors and Actuators A: Physical*, 309:111990, 2020.

[19] Guang Li, Ren Togo, Takahiro Ogawa, and Miki Haseyama. Soft-label anonymous gastric x-ray image distillation. In *ICIP*, 2020.

[20] Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv preprint arXiv:2002.07394*, 2020.

[21] Shikun Li, Xiaobo Xia, Shiming Ge, and Tongliang Liu. Selective-supervised contrastive learning with noisy labels. In *CVPR*, 2022.

[22] Zekai Li, Xinhao Zhong, Samir Khaki, Zhiyuan Liang, Yuhao Zhou, Mingjia Shi, Ziqiao Wang, Xuanlei Zhao, Wangbo Zhao, Ziheng Qin, Mengxuan Wu, Pengfei Zhou, Haonan Wang, David Junhao Zhang, Jia-Wei Liu, Shaobo Wang, Dai Liu, Linfeng Zhang, Guang Li, Kun Wang, Zheng Zhu, Zhiheng Ma, Joey Tianyi Zhou, Jiancheng Lv, Yaochu Jin, Peihao Wang, Kaipeng Zhang, Lingjuan Lyu, Yiran Huang, Zeynep Akata, Zhiwei Deng, Xindi Wu, George Cazenavette, Yuzhang Shang, Justin Cui, Jindong Gu, Qian Zheng, Hao Ye, Shuo Wang, Xiaobo Wang, Yan Yan, Angela Yao, Mike Zheng Shou, Tianlong Chen, Hakan Bilen, Baharan Mirzasoleiman, Manolis Kellis, Konstantinos N. Plataniotis, Zhangyang Wang, Bo Zhao, Yang You, and Kai Wang. Dd-Ranking: Rethinking the evaluation of dataset distillation. *arXiv preprint arXiv:2505.13300*, 2025.

[23] Songhua Liu and Xinchao Wang. Few-shot dataset distillation via translative pre-training. In *ICCV*, 2023.

[24] Yanqing Liu, Jianyang Gu, Kai Wang, Zheng Zhu, Wei Jiang, and Yang You. Dream: Efficient dataset distillation by representative matching. In *ICCV*, 2023.

[25] Yang Lu, Yiliang Zhang, Bo Han, Yiu-ming Cheung, and Hanzi Wang. Label-noise learning with intrinsically long-tailed data. In *ICCV*, 2023.

[26] Sijie Mai, Ying Zeng, and Haifeng Hu. Learning from the global view: Supervised contrastive learning of multimodal representation. *Information Fusion*, 100:101920, 2023.

[27] Cong-Duy Nguyen, Thong Nguyen, Duc Anh Vu, and Luu Anh Tuan. Improving multimodal sentiment analysis: Supervised angular margin-based contrastive learning for enhanced fusion representation. *arXiv preprint arXiv:2312.02227*, 2023.

[28] Timothy Nguyen, Zhourong Chen, and Jaehoon Lee. Dataset meta-learning from kernel ridge-regression. *arXiv preprint arXiv:2011.00050*, 2020.

[29] Curtis G Northcutt, Anish Athalye, and Jonas Mueller. Pervasive label errors in test sets destabilize machine learning benchmarks. *arXiv preprint arXiv:2103.14749*, 2021.

[30] Jihong Ouyang, Chenyang Lu, Bing Wang, and Changchun Li. Supervised contrastive learning with corrected labels for

noisy label learning. *Applied Intelligence*, 53(23):29378–29392, 2023.

[31] Dongmin Park, Seola Choi, Doyoung Kim, Hwanjun Song, and Jae-Gil Lee. Robust data pruning under label noise via maximizing re-labeling accuracy. In *NeurIPS*, 2023.

[32] 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 *CVPR*, 2017.

[33] Douglas A Reynolds et al. Gaussian mixture models. *Encyclopedia of Biometrics*, 741(659-663):3, 2009.

[34] Noveen Sachdeva and Julian McAuley. Data distillation: A survey. *arXiv preprint arXiv:2301.04272*, 2023.

[35] Ahmad Sajedi, Samir Khaki, Ehsan Amjadian, Lucy Z Liu, Yuri A Lawryshyn, and Konstantinos N Plataniotis. Datadam: Efficient dataset distillation with attention matching. In *ICCV*, 2023.

[36] Alexandra M Schnoes, Shoshana D Brown, Igor Dodevski, and Patricia C Babbitt. Annotation error in public databases: misannotation of molecular function in enzyme superfamilies. *PLoS Computational Biology*, 5(12), 2009.

[37] Hooman Sedghamiz, Shivam Raval, Enrico Santus, Tuka Alhanai, and Mohammad Ghassemi. Supcl-seq: Supervised contrastive learning for downstream optimized sequence representations. *arXiv preprint arXiv:2109.07424*, 2021.

[38] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, 2020.

[39] Hwanjun Song, Minseok Kim, and Jae-Gil Lee. Selfie: Refurbishing unclean samples for robust deep learning. In *ICML*, 2019.

[40] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 34(11):8135–8153, 2022.

[41] Ilia Sucholutsky and Matthias Schonlau. Secdd: Efficient and secure method for remotely training neural networks. *arXiv preprint arXiv:2009.09155*, 2020.

[42] Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie. Learning from noisy large-scale datasets with minimal supervision. In *CVPR*, 2017.

[43] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. Dataset distillation. *arXiv preprint arXiv:1811.10959*, 2018.

[44] Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, Gang Niu, and Yang Liu. Learning with noisy labels revisited: A study using real-world human annotations. *arXiv preprint arXiv:2110.12088*, 2021.

[45] Carole-Jean Wu, Ramya Raghavendra, Udit Gupta, Bilge Acun, Newsha Ardalani, Kiwan Maeng, Gloria Chang, Fiona Aga, Jinshi Huang, Charles Bai, et al. Sustainable ai: Environmental implications, challenges and opportunities. In *MLSys*, 2022.

[46] Xiaobo Xia, Bo Han, Yibing Zhan, Jun Yu, Mingming Gong, Chen Gong, and Tongliang Liu. Combating noisy labels with sample selection by mining high-discrepancy examples. In *ICCV*, 2023.

[47] Han Xiao, Huang Xiao, and Claudia Eckert. Adversarial label flips attack on support vector machines. In *ECAI 2012*, pages 870–875. IOS Press, 2012.

[48] Chun Yang, Jianxiao Zou, JianHua Wu, Hongbing Xu, and Shicai Fan. Supervised contrastive learning for recommendation. *Knowledge-Based Systems*, 258:109973, 2022.

[49] Suqin Yuan, Lei Feng, and Tongliang Liu. Early stopping against label noise without validation data. In *ICLR*, 2024.

[50] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[51] Haoyu Zhang, Dingkun Long, Guangwei Xu, Muhua Zhu, Pengjun Xie, Fei Huang, and Ji Wang. Learning with noise: improving distantly-supervised fine-grained entity typing via automatic relabeling. In *IJCAI*, 2021.

[52] Lei Zhang, Jie Zhang, Bowen Lei, Subhabrata Mukherjee, Xiang Pan, Bo Zhao, Caiwen Ding, Yao Li, and Dongkuan Xu. Accelerating dataset distillation via model augmentation. In *CVPR*, 2023.

[53] Bo Zhao and Hakan Bilen. Dataset condensation with differentiable siamese augmentation. In *ICML*, 2021.

[54] Bo Zhao and Hakan Bilen. Dataset condensation with distribution matching. In *WACV*, 2023.

[55] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching. *arXiv preprint arXiv:2006.05929*, 2020.

[56] Ganlong Zhao, Guanbin Li, Yipeng Qin, and Yizhou Yu. Improved distribution matching for dataset condensation. In *CVPR*, 2023.

[57] Yanlin Zhou, George Pu, Xiyao Ma, Xiaolin Li, and Dapeng Wu. Distilled one-shot federated learning. *arXiv preprint arXiv:2009.07999*, 2020.

[58] Yongchao Zhou, Ehsan Nezhadarya, and Jimmy Ba. Dataset distillation using neural feature regression. In *NeurIPS*, 2022.