applied sciences

MDPI

Article

Dual-Training-Based Semi-Supervised Learning with Few Labels

Hao Wu [1], Jun Sun [2] and Qidong Chen [1,*]

[1] School of Internet of Things Engineering, Wuxi University, Wuxi 214105, China; wuhao940917@gmail.com
[2] School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi 214122, China; junsun@jiangnan.edu.cn
* Correspondence: cqd_jnu@hotmail.com

Abstract: The continual expansion in the number of images poses a great challenge for the annotation of the data. Therefore, improving the model performance for image classification with limited labeled data has become an important problem to solve. To address the problem, we propose in this paper a simple and effective dual-training-based semi-supervised learning method for image classification. To enable the model to acquire more valuable information, we propose a dual training approach to enhance model training. Specifically, the model is trained with different augmented data at the same time with soft labels and hard labels, respectively. In addition, we propose a simple and effective weight generation method for generating the weight of samples during training to guide the model training. To further improve the model performance, we employ a projection layer at the end of the network to guide the self-learning of the model by minimizing the distance of features extracted from different layers. Finally, we evaluate the proposed approach on three benchmark image classification datasets. The experimental results demonstrate the effectiveness of our proposed approach.

Keywords: semi-supervised learning; self-supervised learning; image classification; dual training

Citation: Wu, H.; Sun, J.; Chen, Q. Dual-Training-Based Semi-Supervised Learning with Few Labels. *Appl. Sci.* **2024**, *14*, 4993. https://doi.org/10.3390/app14124993

Received: 29 April 2024
Revised: 23 May 2024
Accepted: 5 June 2024
Published: 7 June 2024

1. Introduction

Deep learning has advanced swiftly in recent years. Great breakthroughs caused by deep learning methods have been made in various computer vision tasks, such as classification tasks [1,2], detection tasks [3,4], segmentation tasks [5,6] and so on. However, the success is mainly attributed to huge labeled datasets, such as ImageNet [7]. In addition, annotating such a large dataset is time-consuming and labor-intensive. Hence, it has become a hot topic of research to find how to enhance the performance of the model with only a few labeled data and many unlabeled data. The labeled data are usually annotated manually, while unlabeled data are the data without such annotations. For a few labeled data, researchers have proposed few-shot learning to improve the model performance [8,9]. However, this kind of method ignores the benefits of unlabeled data. Thus, semi-supervised learning methods are proposed to enhance the model performance with few labeled data and many unlabeled data [10–12]. Recently, semi-supervised learning methods have developed rapidly and achieved surprising success for image classification with only a few labeled samples [13–16].

The use of large amounts of unlabeled data is the key to the success of the semi-supervised learning method. In the early research on the semi-supervised learning method, Lee et al. [17] proposed a simple method to make effective use of unlabeled data. They trained a model with existing labeled data first and then generated the pseudo-labels for unlabeled data by the model predictions. Afterward, the model was trained by the unlabeled data with their pseudo-labels. However, due to the limited number of labeled data, the effectiveness of the trained model cannot be guaranteed, and thus the accuracy of generated pseudo-labels is poor, which significantly degrades the model performance. Subsequently, researchers have attempted to utilize consistency regularization to train the model [18,19]. This kind of method assumes that the predictions of the model should be

the same when given the same data under small perturbations. However, training the model by consistency regularization only helps the model to learn itself, but not to learn task-relevant information. Task-relevant information can only be learned from the small amount of available labeled data. As a result, the performance of the models trained by such methods decreases significantly when the amount of labeled data decreases.

To address the above issues, FixMatch [16] combines the advantages of pseudo-labeling methods and consistency-based methods to propose a simple and effective semi-supervised learning method. Specifically, it converts the model prediction of the samples augmented by weak data augmentation (e.g., random cropping and flipping) to one-hot labels. Then, the model is trained with the data augmented by strong data augmentation (e.g., RandAugment [20], CTAugment [21]) with their one-hot labels. Furthermore, the method only selects data with a high value of model prediction to train the model. The method achieved great success for semi-supervised image classification and thus aroused the enthusiasm of researchers for further study. After that, following the architecture of FixMatch, some methods were proposed to improve the performance of the model [22,23]. CoMatch [22] proposes graph-based contrastive learning to impose constraints on features. SimMatch [23] further improves upon FixMatch by combining both semantic similarity and instance similarity. However, the performance of these methods degrades sharply when the amount of labeled data is further reduced [24].

Yang et al. [24] investigated the case when there are only a few labeled data. They proposed a semi-supervised learning method based on interpolated contrastive learning to improve the performance of the model when only two or three labels exist for each category. Hu et al. [25] proposed a patch-mixing contrastive regularization method to ensure that the feature representation is consistent with the task, thus improving the semi-supervised classification performance with only few labeled data. However, these methods are based on FixMatch, which only selects unlabeled data with high confidence for classification training and does not fully utilize the available data.

Therefore, we propose a dual-training-based semi-supervised image classification method in this paper. We first add an extra projection layer at the end of the backbone, as shown in Figure 1. Although our method constructs an extra layer, the number of parameters of the extra layer can be almost negligible. Thus, it does not require more memory of GPU. Produced using WideResNet-28-2 as the backbone, the statistics of the number of parameters for different methods are shown in Table 1.

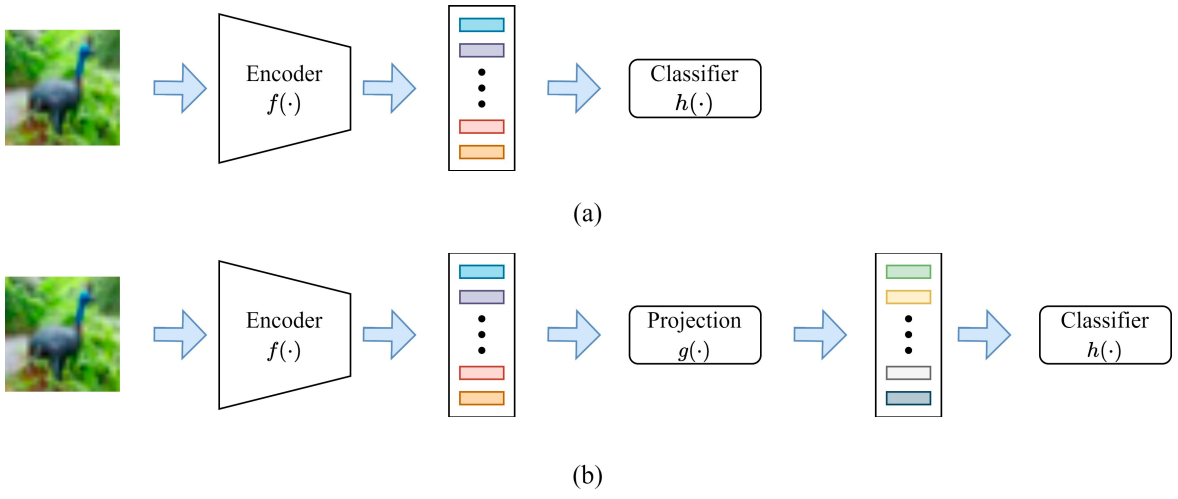

Figure 1. The proposed model architecture. (**a**) backbone; (**b**) our model. Different colors represent different features.

Table 1. The number of parameters for different methods with WideResNet-28-2 as the backbone.

Method	FixMatch [16]	CoMatch [22]	SimMatch [23]	Ours
Parameters(M)	1.4676	1.4924	3.0013	1.4773

Furthermore, we propose a dual training approach that allows the model to learn various useful knowledge simultaneously. Specifically, for unlabeled data, we generate the pseudo-labels with the model prediction of weakly augmented data. After that, we train the model with the data augmented by two different strong data augmentations. The randomness of data augmentation helps the model learn various information simultaneously. In addition, to further increase the difference between the information learned by the model, we respectively utilize the hard labels and soft labels for model training. Hard labels are the one-hot form of model predictions. In addition, hard labels are generated by the same method in FixMatch, and we also utilize the same threshold to select data with high confidence for training, while soft labels generated by the model predictions of weakly augmented data can be seen as hard labels with label smooth. In addition, we use distribution alignment to further improve the soft labels.

To further improve the training of the model with soft-labeled data, we propose a simple and effective weight generation method for generating sample weights. Specifically, for each batch, we normalize the maximum probability of model prediction by the Softmax function to get the weight of each sample. Then we multiply the weight and the number of data so that the sum value of the weights is the same as before.

Inspired by the effectiveness of self-supervised learning [26], we utilize the features extracted from different layers with data augmented by different augmentations to guide the self-learning of the model. By minimizing the cosine distance between these features, the model can learn more useful information by itself.

Finally, we evaluate our method on three benchmark datasets for image classification and verify that the proposed approach can effectively improve the performance of the model for image classification with only few labeled data.

The rest of the paper is organized as follows. The related works of semi-supervised learning and self-supervised learning are described in Section 2. The details of our proposed method are shown in Section 3. The implementation details and the experimental results are described and discussed in Section 4. Finally, the conclusion is presented in Section 5.

2. Related Work

2.1. Semi-Supervised Learning

Since it is a time-consuming and labor-intensive task to annotate a huge number of data, how to fully utilize a small number of labeled data to maintain model performance has become an urgent issue. A semi-supervised learning method has shown its potential to solve the issue. Recently, the most popular methods designed for semi-supervised learning include pseudo-labeling methods [17,27–29] and consistency-based methods [18,19,30]. The purpose of pseudo-labeling methods is to generate pseudo-labels for unlabeled data and then regard them as labeled data to train the model. Therefore, Lee et al. [17] first used the model predictions as pseudo-labels of unlabeled data. Rizve et al. [27] further selected the samples with high confidence to be labeled and introduced complementary labels for unlabeled data with low confidence. Then they further selected samples to train the model by the uncertainty estimation method. Iscen et al. [28] constructed the nearest neighbor graph based on features to annotate unlabeled data. The main idea of consistency-based methods is consistency regularization. Tarvainen et al. [18] built a teacher model by exponential moving average and trained the model to produce the same prediction as that of the teacher model. MixMatch [19] combines the labeled data and unlabeled data by MixUp [31] and then minimizes the difference between the model predictions from data with various augmentations. Miyato et al. [30] perturbed the input data by virtual adversarial loss.

Recently, some pseudo-labeling methods combined the advantages of consistency-based methods and achieved great success for semi-supervised image classification. FixMatch [16] utilizes weakly augmented data to generate pseudo-labels and trains the model by strongly augmented data. It also sets a threshold to select samples that need to be learned. Based on the architecture of FixMatch, CoMatch [22] learns the class probabilities and low-dimensional embeddings of the data. SimMatch [23] further takes semantic similarity and instance similarity into consideration and improves the model performance. Yang et al. [24] introduced interpolated contrastive learning. Hu et al. [25] proposed a patch-mixing contrastive-learning-based method for image classification and achieved impressive results with few labeled data.

2.2. Self-Supervised Learning

Self-supervised learning is one of the effective unsupervised learning methods, which is often seen as the pretraining of the model by exploring the relationships between data or features without the utilization of labels. By training with a self-supervised learning method, the models often have better initial parameters for subsequent specific tasks, such as classification tasks, detection tasks, segmentation tasks and so on. SimCLR [32] constructs positive and negative sample pairs by the data with different augmentation. Specifically, the same data augmented with different data augmentation are considered as a positive sample pair, while different data compose negative sample pairs. The purpose of SimCLR is to reduce the distance between features of positive sample pairs and expand the distance between features of negative sample pairs. BYOL [33] trains the model only with positive samples and abandons the negative sample pairs. SwAV [34] further introduces the clustering method and clusters the data by Sinkhorn–Knopp [35]. It minimizes the difference between the clustering results of the same data with different data augmentation to train the model. SimSiam [26] proposes a simple self-supervised learning method based on a Siamese network, which removes the need for clustering and moving exponential averaging. SimSiam [26] further simplifies the self-supervised learning method while ensuring the effectiveness of the method.

3. Our Approach

In this section, we describe the details of our proposed method. We show the training procedure of our method in Figure 2. We first present the details of our proposed dual training strategy, and then describe the generation of sample weight. Finally, we introduce the cosine-distance-loss-based self-supervised learning method. Algorithm 1 shows the procedure of our method.

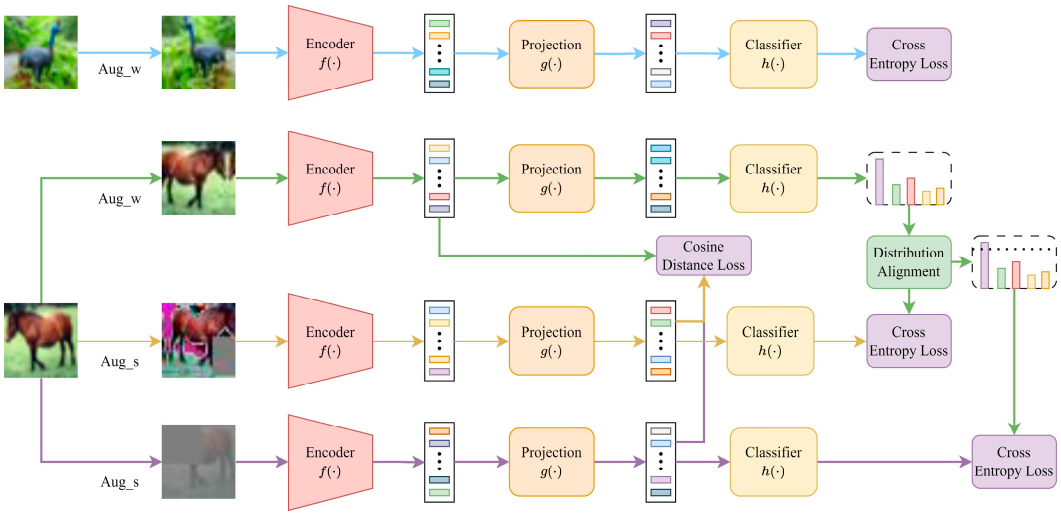

Figure 2. The training procedure of our method. Different colors represent different modules and features.

Algorithm 1 Dual-Training-Based Semi-Supervised Learning.

Input: Labeled dataset $D_l = \left\{(x_i^l, y_i^l)\right\}_{m=1}^{M}$, Unlabeled dataset $D_u = \{x_i^u\}_{n=1}^{N}$
Setting: Learning rate lr, Batch size B, Ratio of unlabeled data μ, Loss weight α, β, γ, Threshold τ, SGD algorithm with momentum, maximum epochs
Initialization: Encoder $f(\cdot)$, Projection $g(\cdot)$, Classifier $h(\cdot)$
for epoch < maximum epochs **do**
  $x_i^{l,w} = Aug\_w(x_i^l)$
  $x_i^{u,w}, x_{i,1}^{u,s}, x_{i,2}^{u,s} = Aug\_w(x_i^u), Aug\_s(x_i^u), Aug\_s(x_i^u)$
  $L_{sup} = L_{cls}(x_i^{l,w}, y_i^l)$
  $pl_i^{soft} = h(g(f(x_i^{u,w})))$
  $pl_i^{soft} = DA(pl_i^{soft})$
  $pl_i^{hard} = argmax(pl_i^{soft})$
  $mask = \mathbb{1}(max(pl_i^{soft}) \geq \tau)$
  $\omega = Softmax(max(pl_i^{soft})) \times B$
  $L_{dual} = \alpha \frac{1}{N} \sum_{i=1}^{N} \omega L_{cls}(x_{i,1}^{u,s}, pl_i^{soft}) + \beta \frac{1}{N} \sum_{i=1}^{N} mask * L_{cls}(x_{i,2}^{u,s}, pl_i^{hard})$
  $L_{dis} = \frac{1}{2}(L_{cos}(g(f(x_{i,1}^{u,s})), f(x_i^{u,w})) + L_{cos}(g(f(x_{i,2}^{u,s})), f(x_i^{u,w})))$
  $L_{total} = L_{sup} + L_{dual} + \gamma L_{dis}$
  Update Encoder $f(\cdot)$, Projection $g(\cdot)$, Classifier $h(\cdot)$
end for
Output: Encoder $f(\cdot)$, Projection $g(\cdot)$, Classifier $h(\cdot)$

3.1. Dual Training Strategy

FixMatch [16] found that training the model with strongly augmented data whose labels are generated by the model prediction of weakly augmented data can achieve great success for image classification. However, it set a threshold to select samples with high confidence and ignore other samples. In addition, the model converges slowly because of the high randomness of data augmentation. A large number of iterations is required for FixMatch to achieve satisfactory results. Therefore, we proposed a novel dual training strategy to fully utilize all data and help the model learn more meaningful information.

Given a labeled dataset $D_l = \left\{(x_i^l, y_i^l)\right\}_{i=1}^{M}$ and an unlabeled dataset $D_u = \{x_i^u\}_{i=1}^{N}$, where M and N respectively represent the number of labeled and unlabeled data, x_i^l and y_i^l are the i^{th} labeled datum and its label. x_i^u is the i^{th} unlabeled datum. For the unlabeled dataset, we first augment the data with weak data augmentation and fit them into the model to obtain the predictions, which are used as the soft labels.

$$pl_i^{soft} = h(g(f(x_i^{u,w}))) \tag{1}$$

Inspired by the effectiveness of distribution alignment [21,36], we also apply the distribution alignment to soft labels to improve their accuracy:

$$pl_i^{soft} = pl_i^{soft} \times h(g(f(x_i^{l,w}))) / \overset{\sim soft}{pl_i} \tag{2}$$

$$pl_i^{soft} = \frac{pl_i^{soft}}{\sum_{j=1}^{N} pl_j^{soft}} \tag{3}$$

where $f(\cdot)$, $g(\cdot)$ and $h(\cdot)$, respectively, mean the encoder, the projection layer and the classifier. $\overset{\sim soft}{pl_i}$ is computed by the moving average of the prediction of unlabeled data.

Next, we can generate hard labels from soft labels:

$$pl_i^{hard} = argmax(pl_i^{soft}) \tag{4}$$

where the $argmax(\cdot)$ function is used to yield one-hot labels from soft labels. Then, the threshold τ is set to select the data with high confidence:

$$mask = \mathbb{1}(max(pl_i^{soft}) \geq \tau) \tag{5}$$

where $\mathbb{1}(\cdot)$ is the mask function to choose the samples whose maximum prediction is higher than threshold τ. After computing the soft and hard labels, we can train the model with unlabeled data by standard cross-entropy loss:

$$L_{soft} = \frac{1}{N}\sum_{i=1}^{N} \omega L_{cls}(x_{i,1}^{u,s}, pl_i^{soft}) \tag{6}$$

$$L_{hard} = \frac{1}{N}\sum_{i=1}^{N} mask L_{cls}(x_{i,2}^{u,s}, pl_i^{hard}) \tag{7}$$

where ω is the weight of each sample, which is described in Section 3.2.

In summary, the total loss of dual training can be written as

$$L_{dual} = \alpha L_{soft} + \beta L_{hard} \tag{8}$$

where α and β are the weights of L_{dual}. With dual training strategy, we train the model with all unlabeled data to fully utilize the existing data. Furthermore, we train the model with samples augmented by different strong augmentation at the same time and the labels of used samples are different (i.e., soft labels and hard labels). Therefore, the model can learn more meaningful information at once, which significantly improves the performance of the model.

3.2. The Generation of Sample Weight

Each sample is weighted equally in the standard categorical cross-entropy loss. However, the semi-supervised learning method often labels unlabeled data by the model trained with the original labeled data. As a result, the accuracy of the model predictions decreases as the amount of labeled data decreases. Hence, training the model with samples with the same weights can make the model learn more incorrect information. Giving different weights to the samples is a simple and effective way to overcome this disadvantage. For instance, MentorNet [37] constructs an additional model to learn the sample weights and train the other model by weighted samples. However, this method needs an additional model. Focal loss [38] adds a scaling factor to the standard categorical cross-entropy loss to control the weight of different categories. However, the focal loss is originally designed for object detection and the weights are mainly for categories. In addition, the focal loss requires artificially set hyperparameters. To address these problems, we proposed a simple way to generate weights for different samples, which is displayed in Figure 3.

We first apply the distribution alignment to the prediction of the model to obtain the soft labels. Then, we select the maximum probability of each label and concatenate them into a vector $V \in \mathbb{R}^{B\times 1}$, where B is the batch size. Afterward, we use the Softmax function to normalize the vector:

$$V = Softmax(max(pl_i^{soft})) \tag{9}$$

As the original sum of sample weights is $B \times 1$ and the sum of the normalized vector is only 1, when we use the normalized vector as the sample weight, it significantly reduces the information learned by the model with each image batch. Therefore, we multiply the normalized vector by the batch size to keep the sum of the sample weights constant. Thus, the weight of samples can be given by

$$\omega = V \times B \tag{10}$$

With the sample weight ω, we can make the model focus more on the samples with high confidence but also learn from the other samples. Thus, it can make full use of all samples.

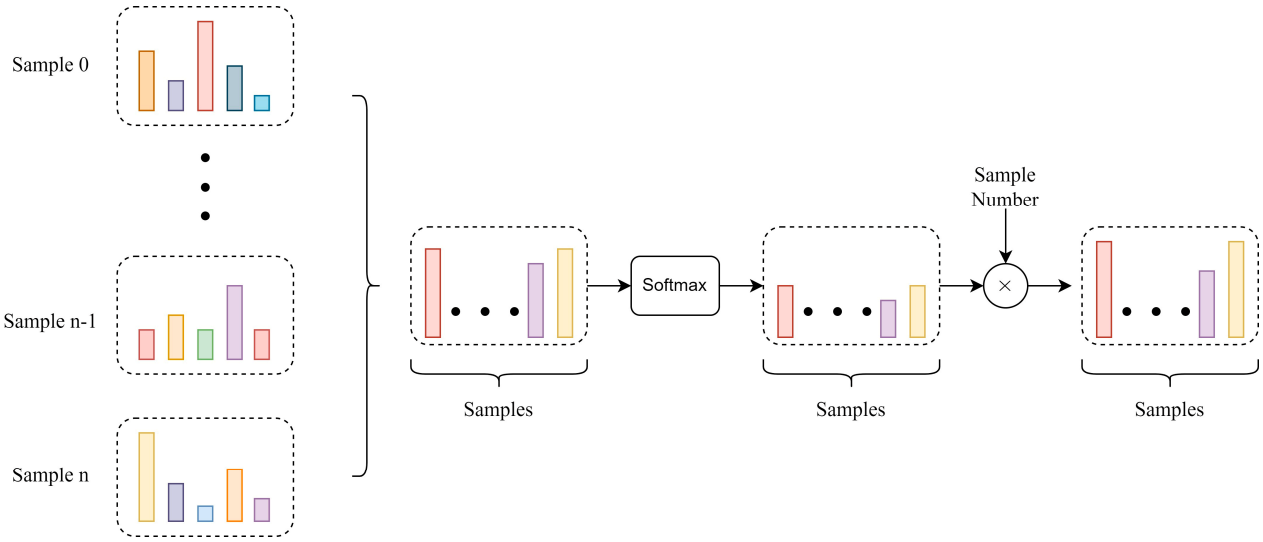

Figure 3. The generation of the sample weight. Different colors represent different dimensions of vectors.

3.3. Self-Supervised Learning Based on Cosine Distance

SimSiam [26] designs a simple and effective self-supervised learning method based on the Siamese network. It can achieve competitive results without a large batch size compared with prior works [32–34]. The main idea of it is that the features extracted by the same sample should be as similar as possible. Inspired by the effectiveness of feature learning, we minimize the feature distance to promote the model learning from itself, which can be seen in Figure 4. Specifically, we regard the features extracted by weakly augmented data from the encoder as the basic features. The features extracted by strongly augmented data from the projection layer are seen as learnable features. We train the model by minimizing the cosine distance between learnable features and basic features. Thus, the loss of self-supervised learning can be written as

$$L_{cos} = \frac{1}{2}(D(f_{l1}, f_b) + D(f_{l2}, f_b)) \tag{11}$$

where

$$f_{l1} = g(f(x_{i,1}^{u,s})) \tag{12}$$

$$f_{l2} = g(f(x_{i,2}^{u,s})) \tag{13}$$

$$f_b = f(x_i^{l,w}) \tag{14}$$

$$D(f_{l1}, f_b) = -\frac{f_{l1} \cdot f_b}{\|f_{l1}\|_2 \cdot \|f_b\|_2} \tag{15}$$

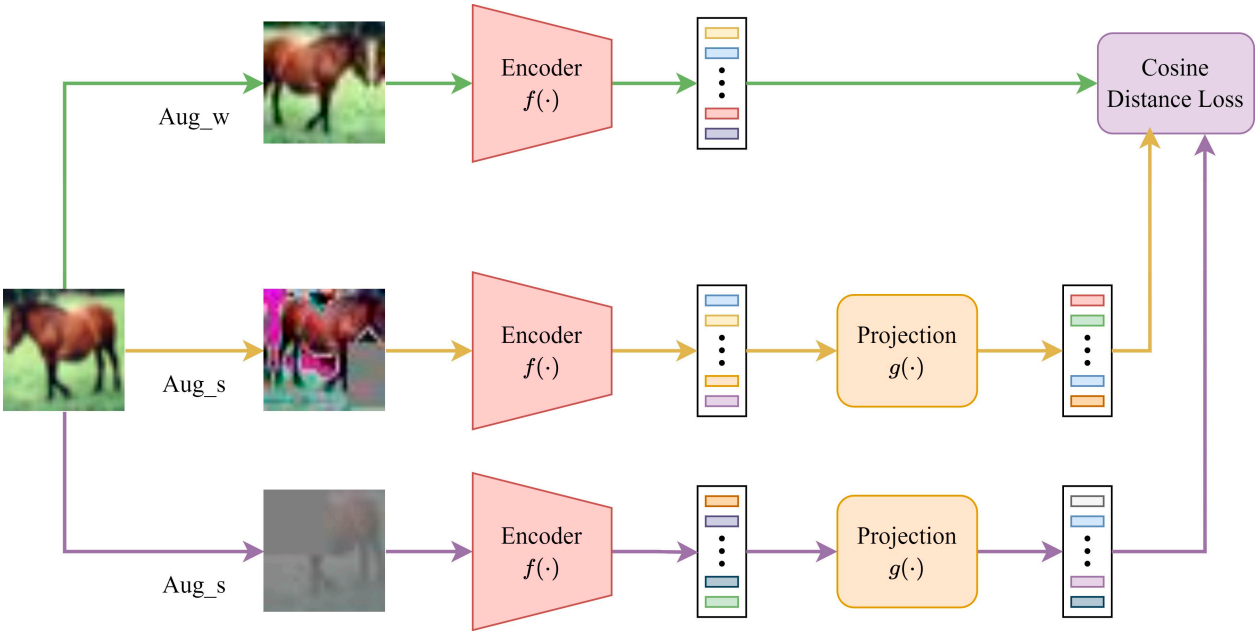

Figure 4. Self-supervised learning based on cosine distance. Different colors represent different modules and features.

4. Experiments

This section describes the information of three classification datasets used in our experiments and presents the implementation details and experimental results to show the performance of the proposed method.

4.1. Datasets

We conducted the experiments on Cifar10, Cifar100 [39] and SVHN [40] datasets for semi-supervised image classification. The information of each dataset is listed in Table 2. The Cifar10 and Cifar100 datasets contain 10 and 100 categories, respectively, and both have 50,000 images for training and 10,000 images for testing. The image size has been standardized to 32 × 32. Following the setting in ICL-SSL [24], for the Cifar10 dataset, we trained the model with three different numbers of labeled data, including 20, 30 and 40. For the Cifar100 dataset, the numbers of labeled data are 200, 400 and 800, respectively. The SVHN dataset contains 99,289 images, each with a size of 32 × 32. The official training and test set include 73,257 and 26,032 images, respectively. We tested our method on the SVHN dataset with 250, 500 and 1000 labeled data, respectively.

Table 2. The information of three benchmark datasets.

Dataset	Image Size	Training Set	Test Set	Categories
Cifar10	32 × 32	50,000	10,000	10
Cifar100	32 × 32	50,000	10,000	100
SVHN	32 × 32	73,257	26,032	10

4.2. Implementation Details

We used the WideResNet-28-2 [41] as our backbone for all datasets. Random flipping and cropping were used as weak augmentation and RandAugment [20] was used as strong augmentation. For a fair comparison, we followed the settings in ICL-SSL [24] and p-Mix [25]. For Cifar10, the batch size was set to 64 and the ratio of unlabeled data was 5. The SGD algorithm was employed for training with the initial learning rate of 0.03. The momentum was 0.9 and the weight decay was 0.0005. The hyperparameters α, β and γ were set to 5, 5 and 0.1, respectively. The number of epochs was 300. For Cifar100, we

set the batch size to 16 and the ratio of unlabeled data was also 5. We employed the SGD algorithm with the same setting as that for Cifar10 to train the model. The initial learning rate was 0.005 and the total epochs were 300. The hyperparameters were set the same as for Cifar10. For the SVHN dataset, the hyperparameters were set $\alpha = 0.5$, $\beta = 5$ and $\gamma = 0.1$, and other settings were the same as for Cifar10.

We trained our method on five runs with different random seeds and reported the mean accuracy and variance as the score of model performance, as the previous work did [25].

4.3. Comparison with State-of-the-Art Methods

We compared our proposed method with several state-of-the-art methods, including π-Model [42], Mean-Teacher [18], MixMatch [19], FixMatch [16], CoMatch [22], SimMatch [23], ICL-SSL [24] and p-Mix [25]. Among them, FixMatch [16] first achieved great success by combining the advantages of pseudo-labeling methods and consistency-based methods. CoMatch [22] and SimMatch [23] introduced the information of categories and features to improve FixMatch. ICL-SSL [24] and p-Mix [25] further considered fewer labeled data and improved the model performance.

We present the comparison results of our method and other compared methods on Cifar10 and Cifar100 with different numbers of labeled data in Table 3. It can be observed that our method outperformed all the other compared methods with different numbers of labeled data. For instance, our method achieved an accuracy of 92.67% with only 20 labeled data on Cifar10, which is 0.72% and 3.94% higher than p-Mix [25] and ICL-SSL [24], respectively. For Cifar100, a more complex dataset with more categories, our method can also achieve the best results, as can be seen by the significant margin between our proposed method and other compared methods. The results demonstrate that our method has the ability to achieve better results with fewer labels, which helps reduce the cost of annotating labels.

Table 3. Comparison with state-of-the-art methods in test accuracy (%) on Cifar10 and Cifar100 datasets with different numbers of labeled data. The bold represents the best result.

Method	Cifar10			Cifar100		
	20 Labels	**30 Labels**	**40 Labels**	**200 Labels**	**400 Labels**	**800 Labels**
π-Model [42]	-	-	-	8.53 ± 0.25	11.67 ± 0.37	17.64 ± 1.0
MeanTeacher [18]	21.79 ± 0.57	24.51 ± 0.35	24.93 ± 0.62	7.11 ± 0.06	11.54 ± 0.28	17.82 ± 0.09
MixMatch [19]	38.51 ± 8.48	50.10 ± 5.81	59.08 ± 3.04	4.55 ± 0.45	17.68 ± 0.07	26.75 ± 1.13
FixMatch [16]	72.63 ± 5.37	86.65 ± 3.56	89.69 ± 4.58	9.31 ± 0.08	24.44 ± 0.35	28.12 ± 0.30
CoMatch [22]	83.43 ± 9.20	88.68 ± 3.79	90.14 ± 2.86	22.39 ± 1.35	29.60 ± 0.88	37.00 ± 0.59
SimMatch [23]	78.13 ± 6.12	90.01 ± 4.15	91.05 ± 3.11	25.43 ± 1.98	38.66 ± 1.61	52.41 ± 0.76
ICL-SSL [24]	88.73 ± 5.69	90.30 ± 3.10	91.78 ± 2.23	14.06 ± 0.52	26.52 ± 1.20	33.81 ± 0.63
p-Mix [25]	91.95 ± 5.95	92.64 ± 1.69	93.65 ± 0.10	25.25 ± 3.28	38.01 ± 2.96	48.24 ± 3.91
Ours	$\mathbf{92.67 \pm 2.59}$	$\mathbf{93.21 \pm 2.45}$	$\mathbf{94.71 \pm 0.26}$	$\mathbf{29.43 \pm 2.02}$	$\mathbf{43.90 \pm 1.49}$	$\mathbf{53.52 \pm 0.54}$

We further investigated the classification accuracy of our method with normal numbers of labeled data on the SVHN dataset. Table 4 illustrates the results of experiments conducted on the SVHN dataset with 250, 500 and 1000 labeled data. As shown in Table 4, our method can still achieve the best result. For instance, our method outperformed the p-Mix [25] with an accuracy of 0.26% on the SVHN dataset with 1000 labeled data, which strongly demonstrates the effectiveness of our method with normal numbers of labeled data.

Table 4. Comparison with state-of-the-art methods in test accuracy (%) on SVHN dataset with 250, 500 and 1000 labeled data. The bold represents the best result.

Method	SVHN		
	250 Labels	**500 Labels**	**1000 Labels**
π-Model [42]	42.66 ± 0.91	53.33 ± 1.39	65.90 ± 0.03
Mean-Teacher [18]	42.70 ± 1.79	55.71 ± 0.53	67.71 ± 1.22
MixMatch [19]	92.12 ± 0.06	94.53 ± 0.43	95.13 ± 0.04
FixMatch [16]	95.45 ± 0.07	95.73 ± 0.15	95.94 ± 0.10
ICL-SSL [24]	95.58 ± 0.14	95.80 ± 0.12	96.05 ± 0.14
p-Mix [25]	97.41 ± 0.02	97.39 ± 0.01	97.45 ± 0.01
Ours	**97.42 ± 0.06**	**97.62 ± 0.06**	**97.71 ± 0.08**

4.4. Ablation Study

This subsection investigates the impact of different components in our method. We first investigated the effectiveness of our proposed dual training strategy for model performance. The experiments were conducted on Cifar100 with 800 labeled data. Since our proposed dual training strategy includes training with soft labels and hard labels, to verify the effectiveness of the dual training strategy, we alternately evaluated the method with only soft labels and only hard labels on Cifar100 with 800 labeled data, which can be seen in Table 5. It should be noted that the dual training strategy trains the model with data from two different kinds of strong data augmentation, while training with single labels only employs data augmented once. In addition, in this paper, training with hard labels means that we only use high-confidence samples selected by threshold. The method achieved an accuracy of 50.96% with only hard labels. The accuracy rose to 51.98% when the model was trained with soft labels instead of hard labels. However, it also showed a significant margin between training with a single kind of label and the dual training strategy. Next, we compared the performance of the model with different dual training strategies. These results are also presented in Table 5, which shows that the model trained by a dual training strategy with both hard and soft labels can reach the highest accuracy of 54.34%, which is 0.55% and 1.97% higher than using only hard labels or only soft labels, respectively. As such, a dual training strategy with both hard and soft labels is beneficial for improving model performance. During the training process, we found that using soft labels to train the model can accelerate the convergence of the model, which can be seen in Figure 5. However, training only with soft labels increases the risk of the model learning incorrect knowledge, resulting in poor performance at the end of the training. Using hard labels makes the model learn less information at the beginning of training but promotes the model's focus on more confident samples, thus reducing the risk of learning incorrect knowledge. Hence, we combined the hard labels and soft labels to train our model and achieved the greatest accuracy.

Table 5. The effectiveness of the dual training strategy on Cifar100 with 800 labeled data. The bold represents the best result.

Method	Test Accuracy (%)
Single hard labels	50.96
Single soft labels	51.98
Dual training with hard labels	53.79
Dual training with soft labels	52.37
Dual training strategy	**54.34**

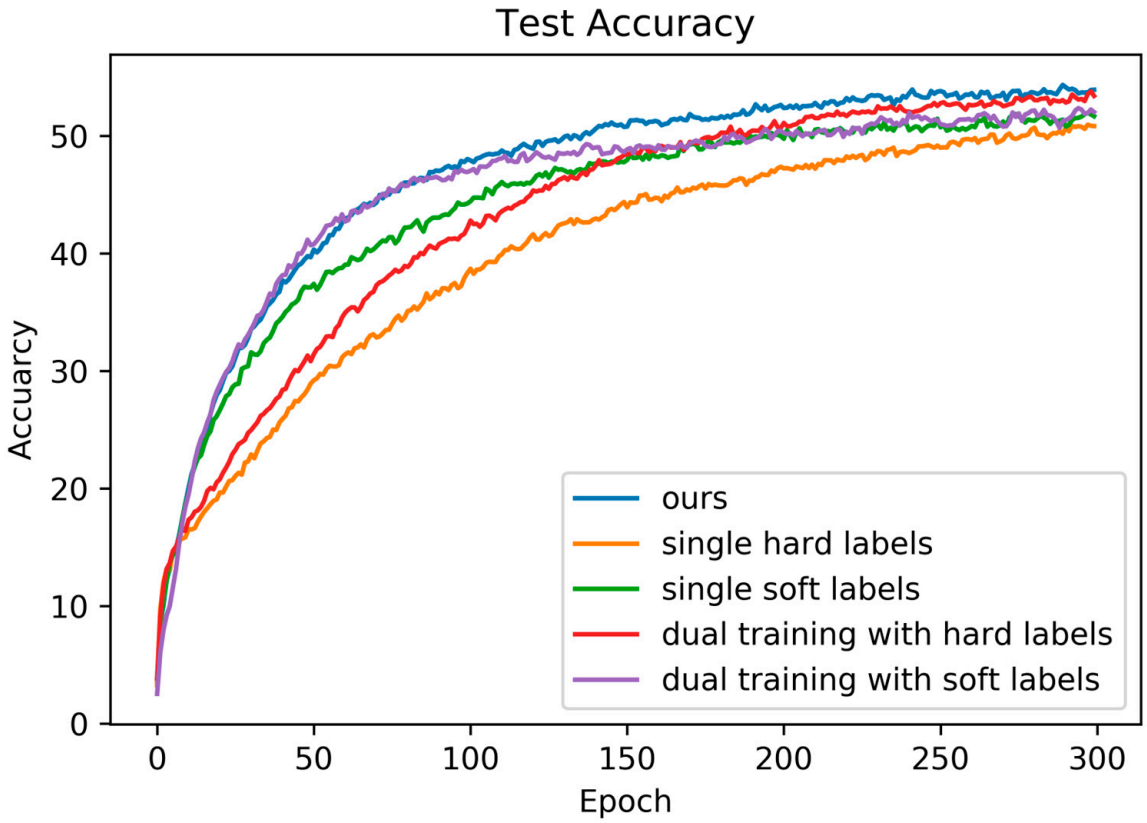

Figure 5. The test accuracy of different methods.

In this study, we employed the distribution alignment to further improve the accuracy of pseudo-labels of unlabeled data. Therefore, in order to investigate the influence of distribution alignment in our method, we tested the method with and without the distribution alignment on Cifar100 with 800 labeled data, the results of which are listed in Table 6. As shown in the table, by simply applying the distribution alignment to our method, the accuracy increased from 51.79% to 54.34%, which significantly demonstrates the importance of distribution alignment. In addition, we proposed a simple weight generation method to improve the training of the model. It is apparent from Table 7 that the model achieved an accuracy of 52.30% without sample weight, which is 2.04% lower than that of the method with sample weight. The experimental results show that distribution alignment and sample weight are beneficial in improving the performance of the model.

Table 6. The effectiveness of distribution alignment on Cifar100 with 800 labeled data. DA: distribution alignment. The bold represents the best result.

Method	Test Accuracy (%)
Ours w/o DA	51.79
Ours with DA	**54.34**

Table 7. The effectiveness of sample weight on Cifar100 with 800 labeled data. SW: sample weight. The bold represents the best result.

Method	Test Accuracy (%)
Ours w/o SW	52.30
Ours with SW	**54.34**

To improve the classification ability of the model and guide the self-training of the model, we minimized the distance between features extracted from different layers by cosine distance loss. We also conducted experiments in Cifar100 with 800 labeled data to investigate the influence of the cosine distance loss. Table 8 illustrates the results of our method with and without cosine distance loss. By adding the cosine distance loss during training, the accuracy rose from 54.04% to 54.34%, which demonstrates the usefulness of the cosine distance loss.

Table 8. The results of our method with and without cosine distance loss on Cifar100 with 800 labeled data. CDL: cosine distance loss. The bold represents the best result.

Method	Test Accuracy (%)
Ours w/o CDL	54.04
Ours with CDL	**54.34**

5. Conclusions

In this paper, we proposed an effective semi-supervised learning method based on dual training for image classification. We improved the model performance with few labels without substantially increasing the number of model parameters. We proposed the dual training strategy, which combines the advantage of soft labels and hard labels, to help the model learn more useful information and fully utilize existing data. In order to prompt the model to focus on the samples with high confidence without ignoring the rest of the samples, we proposed a simple weight generation method to guide the model training. Furthermore, we employed the cosine distance loss based on features to improve the self-learning of the model and enhance the model performance. To evaluate the effectiveness of our proposed method, we conducted experiments on three image classification datasets and compared with other methods. Experimental results demonstrate that our method can work more effectively than other compared methods with few labels. In the future, we will further improve our method by replacing cosine distance and applying a stronger data process.

Author Contributions: Methodology, software, validation, writing—original draft preparation, H.W.; writing—review and editing, J.S.; supervision, Q.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Wuxi University Research Start-up Fund for Introduced Talents (No. 2024r004) and the APC was funded by Wuxi University Research Start-up Fund for Introduced Talents (No. 2024r004).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used in this paper can be found in http://www.cs.toronto.edu/~kriz/cifar.html, accessed on 3 January 2024 and http://ufldl.stanford.edu/housenumbers/, accessed on 3 January 2024.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Arco, J.E.; Ortiz, A.; Ramírez, J.; Martínez-Murcia, F.J.; Zhang, Y.-D.; Górriz, J.M. Uncertainty-driven ensembles of multi-scale deep architectures for image classification. *Inf. Fusion* **2023**, *89*, 53–65. [CrossRef]
2. Sun, L.; Zhao, G.; Zheng, Y.; Wu, Z. Spectral–spatial feature tokenization transformer for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–14. [CrossRef]
3. Shi, S.; Jiang, L.; Deng, J.; Wang, Z.; Guo, C.; Shi, J.; Wang, X.; Li, H. PV-RCNN++: Point-voxel feature set abstraction with local vector representation for 3D object detection. *Int. J. Comput. Vis.* **2023**, *131*, 531–551. [CrossRef]
4. Cheng, G.; Wang, J.; Li, K.; Xie, X.; Lang, C.; Yao, Y.; Han, J. Anchor-free oriented proposal generator for object detection. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–11. [CrossRef]

5. Lin, A.; Chen, B.; Xu, J.; Zhang, Z.; Lu, G.; Zhang, D. Ds-transunet: Dual swin transformer u-net for medical image segmentation. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 4005615. [CrossRef]
6. Ma, F.; Zhang, F.; Xiang, D.; Yin, Q.; Zhou, Y. Fast task-specific region merging for SAR image segmentation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–16. [CrossRef]
7. Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; Li, F.-F. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255. [CrossRef]
8. Snell, J.; Swersky, K.; Zemel, R. Prototypical networks for few-shot learning. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30, pp. 4077–4087. Available online: https://proceedings.neurips.cc/paper_files/paper/2017/file/cb8da6767461f2812ae4290eac7cbc42-Paper.pdf (accessed on 22 May 2024).
9. Wang, Y.; Yao, Q.; Kwok, J.T.; Ni, L.M. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv. (CSUR)* **2020**, *53*, 1–34. [CrossRef]
10. Hua, Z.; Yang, Y. Robust and sparse label propagation for graph-based semi-supervised classification. *Appl. Intell.* **2022**, *52*, 3337–3351. [CrossRef]
11. Liu, P.; Qian, W.; Cao, J.; Xu, D. Semi-supervised medical image classification via increasing prediction diversity. *Appl. Intell.* **2022**, *52*, 10162–10175. [CrossRef]
12. Chen, Y.; Mancini, M.; Zhu, X.; Akata, Z. Semi-supervised and unsupervised deep visual learning: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *46*, 1327–1347. [CrossRef]
13. Xu, H.; Xiao, H.; Hao, H.; Dong, L.; Qiu, X.; Peng, C. Semi-supervised learning with pseudo-negative labels for image classification. *Knowl-Based Syst.* **2023**, *260*, 110166. [CrossRef]
14. Wang, J.; Lukasiewicz, T.; Massiceti, D.; Hu, X.; Pavlovic, V.; Neophytou, A. NP-Match: When Neural Processes meet Semi-Supervised Learning. *arXiv* **2022**, arXiv:2207.01066.
15. Yang, F.; Wu, K.; Zhang, S.; Jiang, G.; Liu, Y.; Zheng, F.; Zhang, W.; Wang, C.; Zeng, L. Class-Aware Contrastive Semi-Supervised Learning. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 14401–14410. [CrossRef]
16. Sohn, K.; Berthelot, D.; Carlini, N.; Zhang, Z.; Zhang, H.; Raffel, C.A.; Cubuk, E.D.; Kurakin, A.; Li, C.L. FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence. In Proceedings of the Advances in Neural Information Processing Systems, Virtual, 6–12 December 2020; Curran Associates, Inc.: New York, NY, USA, 2020; pp. 596–608. Available online: https://proceedings.neurips.cc/paper_files/paper/2020/file/06964dce9addb1c5cb5d6e3d9838f733-Paper.pdf (accessed on 10 January 2024).
17. Lee, D.-H. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In Proceedings of the Workshop on Challenges in Representation Learning (ICML), Atlanta, GA, USA, 16–21 June 2013; p. 896. Available online: https://ui.adsabs.harvard.edu/abs/2022arXiv220110836G (accessed on 10 January 2024).
18. Tarvainen, A.; Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *arXiv* **2017**, arXiv:1703.01780.
19. Berthelot, D.; Carlini, N.; Goodfellow, I.; Papernot, N.; Oliver, A.; Raffel, C.A. MixMatch: A Holistic Approach to Semi-Supervised Learning. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Curran Associates, Inc.: New York, NY, USA, 2019. Available online: https://proceedings.neurips.cc/paper/2019/file/1cd138d0499a68f4bb72bee04bbec2d7-Paper.pdf (accessed on 10 January 2024).
20. Cubuk, E.D.; Zoph, B.; Shlens, J.; Le, Q.V. Randaugment: Practical automated data augmentation with a reduced search space. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Virtual, 6–12 December 2020; pp. 702–703. [CrossRef]
21. Berthelot, D.; Carlini, N.; Cubuk, E.D.; Kurakin, A.; Sohn, K.; Zhang, H.; Raffel, C. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv* **2019**, arXiv:1911.09785.
22. Li, J.; Xiong, C.; Hoi, S.C.H. Comatch: Semi-supervised learning with contrastive graph regularization. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 9475–9484. [CrossRef]
23. Zheng, M.; You, S.; Huang, L.; Wang, F.; Qian, C.; Xu, C. Simmatch: Semi-supervised learning with similarity matching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 14471–14481. [CrossRef]
24. Yang, X.; Hu, X.; Zhou, S.; Liu, X.; Zhu, E. Interpolation-based contrastive learning for few-label semi-supervised learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *35*, 2054–2065. [CrossRef] [PubMed]
25. Hu, X.; Xu, X.; Zeng, Y.; Yang, X. Patch-Mixing Contrastive Regularization for Few-Label Semi-Supervised Learning. *IEEE Trans. Artif. Intell.* **2023**, *5*, 384–397. [CrossRef]
26. Chen, X.; He, K. Exploring simple siamese representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 15750–15758. [CrossRef]
27. Rizve, M.N.; Duarte, K.; Rawat, Y.S.; Shah, M. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *arXiv* **2021**, arXiv:2101.06329.
28. Iscen, A.; Tolias, G.; Avrithis, Y.; Chum, O. Label propagation for deep semi-supervised learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5070–5079. [CrossRef]

29. Cascante-Bonilla, P.; Tan, F.; Qi, Y.; Ordonez, V. Curriculum labeling: Revisiting pseudo-labeling for semi-supervised learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; Volume 35, pp. 6912–6920. [CrossRef]
30. Miyato, T.; Maeda, S.; Koyama, M.; Ishii, S. Virtual adversarial training: A regularization method for supervised and semi-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 1979–1993. [CrossRef] [PubMed]
31. Zhang, H.; Cisse, M.; Dauphin, Y.N.; Lopez-Paz, D. mixup: Beyond empirical risk minimization. *arXiv* **2017**, arXiv:1710.09412.
32. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A simple framework for contrastive learning of visual representations. In Proceedings of the International Conference on Machine Learning (PMLR), Virtual, 13–18 July 2020; pp. 1597–1607. Available online: http://proceedings.mlr.press/v119/chen20j/chen20j.pdf (accessed on 10 January 2024).
33. Grill, J.B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P.; Buchatskaya, E.; Doersch, C.; Avila Pires, B.; Guo, Z.; Gheshlaghi Azar, M.; et al. Bootstrap your own latent-a new approach to self-supervised learning. In Proceedings of the Advances in neural Information Processing Systems, Virtual, 6–12 December 2020; Volume 33, pp. 21271–21284. Available online: https://proceedings.neurips.cc/paper_files/paper/2020/file/f3ada80d5c4ee70142b17b8192b2958e-Paper.pdf (accessed on 10 January 2024).
34. Caron, M.; Misra, I.; Mairal, J.; Goyal, P.; Bojanowski, P.; Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. In Proceedings of the Advances in Neural Information Processing Systems, Virtual, 6–12 December 2020; Volume 33, pp. 9912–9924. Available online: https://proceedings.neurips.cc/paper/2020/file/70feb62b69f16e0238f741fab228fec2-Paper.pdf (accessed on 10 January 2024).
35. Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; Volume 26. Available online: https://proceedings.neurips.cc/paper/2013/file/af21d0c97db2e27e13572cbf59eb343d-Paper.pdf (accessed on 10 January 2024).
36. Wang, Y.; Chen, H.; Fan, Y.; Sun, W.; Tao, R.; Hou, W.; Wang, R.; Yang, L.; Zhou, Z.; Guo, L.Z.; et al. Usb: A unified semi-supervised learning benchmark for classification. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 28 November–9 December 2022; Volume 35, pp. 3938–3961. Available online: https://proceedings.neurips.cc/paper_files/paper/2022/file/190dd6a5735822f05646dc27decff19b-Paper-Datasets_and_Benchmarks.pdf (accessed on 10 January 2024).
37. Jiang, L.; Zhou, Z.; Leung, T.; Li, J.; Li, F.-F. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In Proceedings of the International Conference on Machine Learning (PMLR), Stockholm, Sweden, 10–15 July 2018; pp. 2304–2313. Available online: https://proceedings.mlr.press/v80/jiang18c.html (accessed on 10 January 2024).
38. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988. [CrossRef]
39. Krizhevsky, A. Learning Multiple Layers of Features from Tiny Images. Technical Report. University of Toronto. 2009. Available online: http://www.cs.utoronto.ca/~kriz/learning-features-2009-TR.pdf (accessed on 10 January 2024).
40. Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; Ng, A.Y. Reading Digits in Natural Images with Unsupervised Feature Learning. 2011. Available online: http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf (accessed on 10 January 2024).
41. Zagoruyko, S.; Komodakis, N. Wide residual networks. *arXiv* **2016**, arXiv:1605.07146.
42. Laine, S.; Aila, T. Temporal ensembling for semi-supervised learning. *arXiv* **2016**, arXiv:1610.02242.