algorithms

MDPI

Article

Multi-Augmentation-Based Contrastive Learning for Semi-Supervised Learning

Jie Wang [1], Jie Yang [1,*], Jiafan He [2] and Dongliang Peng [1]

[1] Artificial Intelligence Institute, Hangzhou Dianzi University, Hangzhou 310018, China; wjlylw@hdu.edu.cn (J.W.); dlpeng@hdu.edu.cn (D.P.)
[2] Science and Technology on Information Systems Engineering Laboratory, Nanjing 210014, China; fangf@hdu.edu.cn
* Correspondence: yangjie08@hdu.edu.cn

Abstract: Semi-supervised learning has been proven to be effective in utilizing unlabeled samples to mitigate the problem of limited labeled data. Traditional semi-supervised learning methods generate pseudo-labels for unlabeled samples and train the classifier using both labeled and pseudo-labeled samples. However, in data-scarce scenarios, reliance on labeled samples for initial classifier generation can degrade performance. Methods based on consistency regularization have shown promising results by encouraging consistent outputs for different semantic variations of the same sample obtained through diverse augmentation techniques. However, existing methods typically utilize only weak and strong augmentation variants, limiting information extraction. Therefore, a multi-augmentation contrastive semi-supervised learning method (MAC-SSL) is proposed. MAC-SSL introduces moderate augmentation, combining outputs from moderately and weakly augmented unlabeled images to generate pseudo-labels. Cross-entropy loss ensures consistency between strongly augmented image outputs and pseudo-labels. Furthermore, the MixUP is adopted to blend outputs from labeled and unlabeled images, enhancing consistency between re-augmented outputs and new pseudo-labels. The proposed method achieves a state-of-the-art performance (accuracy) through extensive experiments conducted on multiple datasets with varying numbers of labeled samples. Ablation studies further investigate each component's significance.

Keywords: contrastive learning; multi-augmentation-based method; semi-supervised learning (SSL)

Citation: Wang, J.; Yang, J.; He, J.; Peng, D. Multi-Augmentation-Based Contrastive Learning for Semi-Supervised Learning. *Algorithms* **2024**, *17*, 91. https://doi.org/10.3390/a17030091

Academic Editors: Mario Rosario Guarracino, Laura Antonelli and Pietro Hiram Guzzi

Received: 12 January 2024
Revised: 11 February 2024
Accepted: 16 February 2024
Published: 20 February 2024

1. Introduction

In recent years, deep learning has rapidly advanced and achieved remarkable results in various fields, such as image classification [1,2], object detection [3,4], clustering [5,6], semantic segmentation [7,8], and more. However, the success of deep learning is heavily reliant on large-scale, high-quality labeled datasets [9].

However, collecting labeled data can be expensive and time-consuming, especially when expert annotation is required, which is unaffordable for the countless everyday learning demands in modern society. Semi-supervised learning [10,11] addresses the scarcity of labeled samples by leveraging a combination of a small labeled dataset and a substantial amount of unlabeled data for model training, thus alleviating the dependency on extensive labeled datasets. This has led to a plethora of SSL methods designed for various fields [12–19]. Traditional semi-supervised learning methods involve training models to predict artificial labels for unlabeled images, which are then incorporated as supplementary inputs during training. However, these approaches, such as the pseudo-labeling method [20,21] (also known as self-training [22–26]), face limitations due to their reliance on initial training with labeled samples prior to generating pseudo-labels for unlabeled samples, resulting in reduced effectiveness when labeled data are scarce.

Consistency regularization [27–30] -based semi-supervised learning methods (also commonly referred to as contrastive learning [31–34]) tackle this challenge by treating both

labeled and unlabeled input images, along with their augmented versions, as positive pairs. These consistency regularization-based methods in deep learning operate under the assumption that, even after applying data augmentation [35–37], the classifier should maintain consistent class probabilities for unlabeled samples, implying that the semantic content remains unchanged. The augmented versions of an input image should exhibit greater similarity to the original image compared to other unrelated images. To adhere to this assumption, researchers introduce perturbations to the input samples through data augmentation, thereby generating augmented samples that are similar to the original data.

While the aforementioned algorithms have significantly contributed to improving learning accuracy in situations where labeled data are limited, they exhibit a notable decline in performance when confronted with a scarcity of labeled samples. This decline can be primarily attributed to the inability to fully leverage the informative content present in the unlabeled data, leading to an overreliance on the limited labeled samples.

Different from approaches that typically enforce consistency between the model outputs of strongly augmented unlabeled images and weakly augmented unlabeled images (using pseudo-labels or soft labels), MAC-SSL introduces a moderate augmentation step, where the model outputs of moderately augmented unlabeled images collaborate with the model outputs of weakly augmented images to derive pseudo-labels for the unlabeled images. Subsequently, the consistency (contrastive loss) between the model outputs of strongly augmented images and the pseudo-labels is enforced. Furthermore, inspired by MixMatch [33], the MixUP [38] is adopted to combine the three different outputs of unlabeled images and the outputs of labeled images to obtain further augmentation, which ensures consistency (unsupervised loss) between the model outputs of the re-augmented images and the newly generated pseudo-labels derived from the mixed labels or pseudo-labels.

The main contributions of this article are summarized as follows:

(1) A novel moderate augmentation technique is introduced, which is incorporated into two distinctive losses.
(2) Numerous experiments demonstrate that MAC-SSL achieves state-of-the-art (SOTA) results across all standard benchmark datasets (Section 3.3).
(3) The conducted ablation experiments illustrate the excellent performance of MAC-SSL (Section 3.4).

2. Method

This section provides a detailed introduction to the proposed MAC-SSL algorithm. For an L-class classification problem, we let $\mathcal{X} = ((x_b, q_b); b \in (1, \ldots, B))$ represent a batch of B labeled examples with L classes where x_b is the training sample and q_b is the corresponding one-hot encoded label, and we let $\mathcal{U} = (u_b; b \in (1, \ldots, \mu B))$ represent a batch of μB unlabeled examples, where μ is a hyperparameter that determines the relative sizes of $\mathcal{X}$ and $\mathcal{U}$.

2.1. Data Augmentation

Data augmentation is performed on both labeled and unlabeled data. There are three different levels of augmentation strategies used, and in increasing order of intensity, they are weak augmentation, moderate augmentation, and strong augmentation. Weak augmentation refers to a standard flip-and-shift augmentation strategy. Specifically, it randomly flips an image horizontally with a 50% probability and randomly translates the image by 12.5% in the horizontal or vertical direction. Moderate augmentation involves applying Augmix [39] to weakly augmented samples, following the configuration specified in Augmix [39] and Augmix's pseudocode visible Algorithm 1. Strong augmentation applies RandAugment [40] to weakly augmented samples, following the configuration specified in RandAugment [40].

For the labeled data batch $\mathcal{X}$, a "weak" augmentation strategy is employed. For the unlabeled data batch $\mathcal{U}$, a combination of "weak", "moderate", and "strong" augmentation

techniques is employed. The weak augmentation strategy is identical to that employed for $\mathcal{X}$.

By employing the weak, moderate, and strong augmentation strategies, the following augmented data batches are generated: the weakly augmented labeled data batch $\mathcal{X}'$, the weakly augmented unlabeled data batch $\mathcal{U}'_w$, the moderately augmented unlabeled data batch $\mathcal{U}'_m$, and the strongly augmented unlabeled data batch $\mathcal{U}'_s$.

The data augmentation process described above is illustrated in Figure 1.

Algorithm 1 AugMix

1: Input: Original image x_{orig}, Operations $\mathcal{O} = rotate, \ldots, posterize$.
2: **function** AugmentAndMix $(x_{orig}, k = 3, \alpha = 1)$
3: Fill x_{aug} with zeros of same shape as x_{orig}
4: Sample mixing weights $(w_1, w_2, \ldots, w_k) \sim$Dirichlet $(\alpha, \ldots, \alpha)$ ▷ Dirichlet denotes Dirichlet distribution
5: **for** $i = 1$ to k do
6: $op_1, op_2, op_3 \sim \mathcal{O}$ ▷ Sample augmentation operation
7: Compose operations with varying depth $op_{12} = op_2 \circ op_1, op_{123} = op_3 \circ op_{12}$ ▷ $\circ$ denotes composition of operations
8: Sample uniformly from one of these operations $chain \sim \{op_1, op_{12}, op_{123}\}$
9: $x_{aug} += w_i \cdot chain(x_{orig})$ ▷ Addition is elementwise, $\cdot$ denotes that attach weights to each augmentation operation.
10: **end for** ▷ Completion of the augmetation process
11: Sample weight $m \sim$ Beta(α, α)
12: Interpolation with rule $x_{augmix} = mx_{orig} + (1 - m)x_{aug}$ ▷ Completion of the mix process
13: **return** x_{augmix}
14: **end function**
15: $x_{am} =$ AugmentAndMix $(x_{orig}, 3, 1)$ ▷ x_{am} is stochastically generated by the function AugmentAndMix $(x_{orig}, k = 3, \alpha = 1)$.
16: **return** x_{am}.

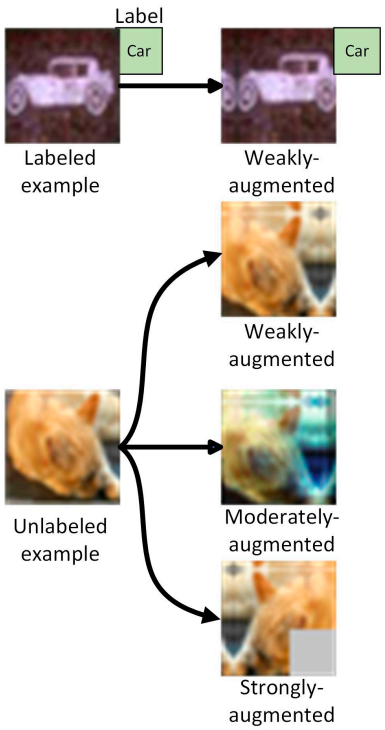


Figure 1. Process of the data augmentation.

2.2. Pseudo-Labels Generating

The process of generating pseudo-labels is depicted in Figure 2. MAC-SSL leverages the model outputs $p_{wb}^{u\prime} = p(q|u'_{wb})$ and $p_{mb}^{u\prime} = p(q|u'_{mb})$ obtained from its weakly augmented and moderately augmented versions u'_{wb} and u'_{mb}, respectively, to compute an average prediction p_b^u for each unlabeled sample u_b in $\mathcal{U}$. It is worth noting that $p(q|u)$ represents the model's output for u. Subsequently, a "Sharpening" [33] operation is applied to p_b^u to generate the pseudo-label q_b^u for u_b. The corresponding computational steps are described as follows:

$$p_b^u = \frac{1}{2}(p_{wb}^{u\prime} + p_{mb}^{u\prime}) \text{ and} \tag{1}$$

$$q_b^u = \text{Sharpen}(p_b^u, T), \tag{2}$$

where the Sharpening operation is defined as follows:

$$\text{Sharpen}(p, T)_i := \frac{p_i^{\frac{1}{T}}}{\sum_{j=1}^{L} p_j^{\frac{1}{T}}}, \tag{3}$$

where p represents an input categorical distribution (specifically in MAC-SSL, p is the average class prediction generated from the model outputs $p_{wb}^{u\prime}$ and $p_{mb}^{u\prime}$, denoted as p_b^u) and T is a hyperparameter that signifies the "temperature [41]" of the class distribution. As $T \to 0$, the output of $\text{Sharpen}(p, T)$ will approach a "one-hot" distribution. Since $q_b^u = \text{Sharpen}(p_b^u, T)$ will be used as a target for the model's prediction for augmentations of u_b, lowering the temperature encourages the model to produce lower-entropy predictions. T is set to a small value in this paper.

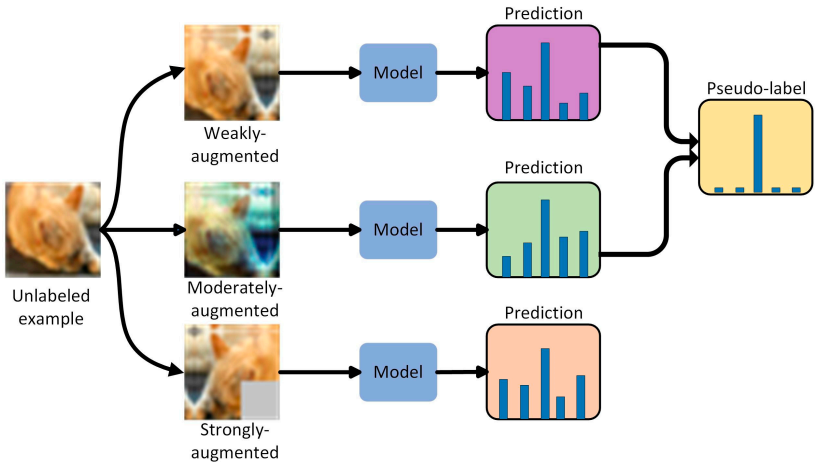


Figure 2. Process of pseudo-label generating.

2.3. MixUP

MixUP is employed for semi-supervised learning, distinguishing itself from prior approaches that solely mix images. MAC-SSL involves mixing both labeled samples with ground-true labels and unlabeled samples with pseudo-labels (generated as described in Section 2.2). For a pair of samples with their respective label predictions (x_1, q_1) and (x_2, q_2), when x_1 is a labeled sample, q_1 represents the true label. When x_1 is an unlabeled sample, q_1 represents the generated pseudo-label. The same applies to x_2 and q_2. The MixUP technique employed can be described as follows:

$$\lambda \sim \text{Beta}(\alpha, \alpha), \tag{4}$$

$$\lambda' = \max(\lambda, 1 - \lambda), \tag{5}$$

$$(x'_1, q'_1) = \lambda'(x_1, q_1) + (1 - \lambda')(x_2, q_2), \text{ and} \tag{6}$$

$$(x'_2, q'_2) = \lambda'(x_2, q_2) + (1 - \lambda')(x_1, q_1), \tag{7}$$

where (x'_1, q'_1) represents the new sample generated by mixing with x_1 as the base (λ' is a value close to 1), (x'_2, q'_2) represents the new sample generated by mixing with x_2 as the base, and α is a hyperparameter. The specific calculation steps for Equation (6) are as follows:

$$x'_1 = \lambda' x_1 + (1 - \lambda')x_2 \text{ and} \tag{8}$$

$$q'_1 = \lambda' q_1 + (1 - \lambda')q_2, \tag{9}$$

and the specific calculation steps for Equation (7) are as follows:

$$x'_2 = \lambda' x_2 + (1 - \lambda')x_1 \text{ and} \tag{10}$$

$$q'_2 = \lambda' q_2 + (1 - \lambda')q_1. \tag{11}$$

The mixing process in the algorithm is presented in line 15 of Algorithm 2.

Algorithm 2 MAC-SSL

1: Input: Batch of labeled examples $\mathcal{X} = ((x_b, q_b); b \in (1, \ldots, B))$, batch of unlabeled examples $\mathcal{U} = (u_b; b \in (1, \ldots, \mu B))$, ratio of sample size μ, sharpening temperature T, Beta distribution parameter α for MixUp, unlabeled loss weight λ_u, contrastive loss weight λ_c.
2: **for** $b = 1$ to B
3: $x'_b =$ Weak-Augment (x_b)
4: **end for**
5: **for** $b = 1$ to μB
6: $u'_{wb} =$ Weak-Augment (u_b), $u'_{mb} =$ Moderate-Augment (u_b), $u'_{sb} =$ Strong-Augment (u_b)
7: $p^{u\prime}_{wb}, p^{u\prime}_{mb}, p^{u\prime}_{sb} = p(q|u'_{wb}), p(q|u'_{mb}), p(q|u'_{sb})$
8: $p^u_b = \frac{1}{2}(p^{u\prime}_{wb} + p^{u\prime}_{mb})$ ▷ Compute average predictions of u'_{wb} and u'_{mb}
9: $q^u_b = Sharpen(p^u_b, T)$ ▷ Apply temperature sharpening to the average prediction (see Equation (3))
10: **end for**
11: $\mathcal{X}' = ((x'_b, q_b); b \in (1, \ldots, B))$,
12: $\mathcal{U}'_w = (u'_{wb}; b \in (1, \ldots, \mu B)), \mathcal{U}'_m = (u'_{mb}; b \in (1, \ldots, \mu B)), \mathcal{U}'_s = (u'_{sb}; b \in (1, \ldots, \mu B))$
13: $\mathcal{W} =$ Shuffle$(Concat(\mathcal{X}', \mathcal{U}'_w, \mathcal{U}'_m, \mathcal{U}'_s))$ ▷ Combine and shuffle labeled and unlabeled data
14: $\mathcal{X}'' = (\text{MixUp}(\mathcal{X}', \mathcal{W}_i); i \in (1, \ldots, B))$
▷ see Section 2.3
15: $\mathcal{U}''_w = (\text{MixUp}(\mathcal{U}'_w, \mathcal{W}_{i+B}); i \in (1, \ldots, \mu B)), \mathcal{U}''_m = (\text{MixUp}(\mathcal{U}'_m, \mathcal{W}_{i+B+\mu B}); i \in (1, \ldots, \mu B))$,
$\mathcal{U}''_s = (\text{MixUp}(\mathcal{U}'_s, \mathcal{W}_{i+B+2\mu B}); i \in (1, \ldots, \mu B))$
16: $\mathcal{L}_x = \frac{1}{B} \sum_{b=1}^{B} H(q_b, p(q|x_b))$
▷ Equation (12)
17: $\mathcal{L}_{xm} = \frac{1}{B} \sum_{b=1}^{B} H(q'_b, p(q|x''_b))$
▷ Equation (13)
18: $\mathcal{L}_u = \frac{1}{\mu B} \sum_{b=1}^{\mu B} (\|q^{u\prime}_b - p(q \mid u''_{wb})\|_2^2 + \|q^{u\prime}_b - p(q \mid u''_{mb})\|_2^2 + \|q^{u\prime}_b - p(q \mid u''_{sb})\|_2^2)$
▷ Equation (14)
19: $\mathcal{L}_c = \frac{1}{\mu B} \sum_{b=1}^{\mu B} f(\max(p^u_b) \geq \tau) H(p^u_b, p(q \mid u'_{sb}))$
▷ Equation (15)
20: $\mathcal{L} = \mathcal{L}_x + \mathcal{L}_{xm} + \lambda_u \mathcal{L}_u + \lambda_c \mathcal{L}_c$
▷ Equation (16)
21: return $\mathcal{L}$

2.4. The Proposed MAC-SSL

MAC-SSL generates one batch of augmented labeled data, denoted as $\mathcal{X}'$, and three batches of augmented unlabeled data, where $\mathcal{U}'_w$, $\mathcal{U}'_m$, and $\mathcal{U}'_s$. $\mathcal{X}'$ represent the labeled data batch after weak augmentation and $\mathcal{U}'_w$, $\mathcal{U}'_m$, and $\mathcal{U}'_s$ represent the unlabeled data batches after weak, moderate, and strong augmentations, respectively. The average prediction distribution p_b^u is computed using the model outputs $p_{wb}^{u\prime}$ and $p_{mb}^{u\prime}$ from $\mathcal{U}'_w$ and $\mathcal{U}'_m$, respectively (see Equation (1)), which are then used to generate the pseudo-labels $q_b^{u\prime}$ for the unlabeled data batch (see Equation (2)). Subsequently, $\mathcal{X}'$ and $\mathcal{U}'_w$, $\mathcal{U}'_m$, and $\mathcal{U}'_s$ are mixed to generate the mixed weakly augmented labeled data batch $\mathcal{X}''$ and the mixed weakly augmented unlabeled data batches $\mathcal{U}''_w$, the mixed moderately augmented unlabeled data batch $\mathcal{U}''_m$, and the mixed strongly augmented unlabeled data batch $\mathcal{U}''_s$, respectively. Likewise, the ground-true label q_b and pseudo-label q_b^u are also mixed to obtain the corresponding pseudo-labels q'_b and $q_b^{u\prime}$ for $\mathcal{X}''$ and $\mathcal{U}''_w$, $\mathcal{U}''_m$, and $\mathcal{U}''_s$, respectively (see Section 2.3). The complete MAC-SSL framework is illustrated in Figure 3, and a flowchart of the MAC-SSL process is presented in Figure 4.

The loss function of MAC-SSL consists of the following four components: supervised classification loss $\mathcal{L}_x$, mixed supervised classification loss $\mathcal{L}_{xm}$, unsupervised classification loss $\mathcal{L}_u$, and contrastive loss $\mathcal{L}_c$.

$\mathcal{L}_x$ is the supervised classification loss on $\mathcal{X}$, which is defined as the cross-entropy between the ground-true label and the model prediction, as follows:

$$\mathcal{L}_x = \frac{1}{B}\sum_{b=1}^{B} H(q_b, p(q|x_b)), \tag{12}$$

where $p(q|x_b)$ is the model's prediction for $x_b \in \mathcal{X}$ and H denotes the cross-entropy between q_b and $p(q|x_b)$.

Regarding $\mathcal{L}_{xm}$, it is the supervised classification loss on $\mathcal{X}''$. It is defined as the cross-entropy between the pseudo-label and the model prediction, as follows:

$$\mathcal{L}_{xm} = \frac{1}{B}\sum_{b=1}^{B} H(q'_b, p(q|x''_b)), \tag{13}$$

where q'_b is the pseudo-label, $p(q|x''_b)$ is the model's prediction for $x''_b \in \mathcal{X}''$, and H denotes the cross-entropy between q'_b and $p(q|x''_b)$.

The unsupervised classification loss, $\mathcal{L}_u$, is defined on $\mathcal{U}''_w$, $\mathcal{U}''_m$, and $\mathcal{U}''_s$. It is computed as the mean squared error between the pseudo-label $q_b^{u\prime}$ and the three model predictions $p(q|u''_{wb})$, $p(q|u''_{mb})$, and $p(q|u''_{sb})$ as follows:

$$\mathcal{L}_u = \frac{1}{\mu B}\sum_{b=1}^{\mu B} (\|q_b^{u\prime} - p(q \mid u''_{wb})\|_2^2 + \|q_b^{u\prime} - p(q \mid u''_{mb})\|_2^2 + \|q_b^{u\prime} - p(q \mid u''_{sb})\|_2^2), \tag{14}$$

where u''_{wb}, u''_{mb}, $u''_{sb} \in \mathcal{U}''_w$, $\mathcal{U}''_m$, and $\mathcal{U}''_s$ and $p(q|u''_{wb})$, $p(q|u''_{mb})$, and $p(q|u''_{sb})$ are the model's predictions for u''_{wb}, u''_{mb}, and u''_{sb}, respectively.

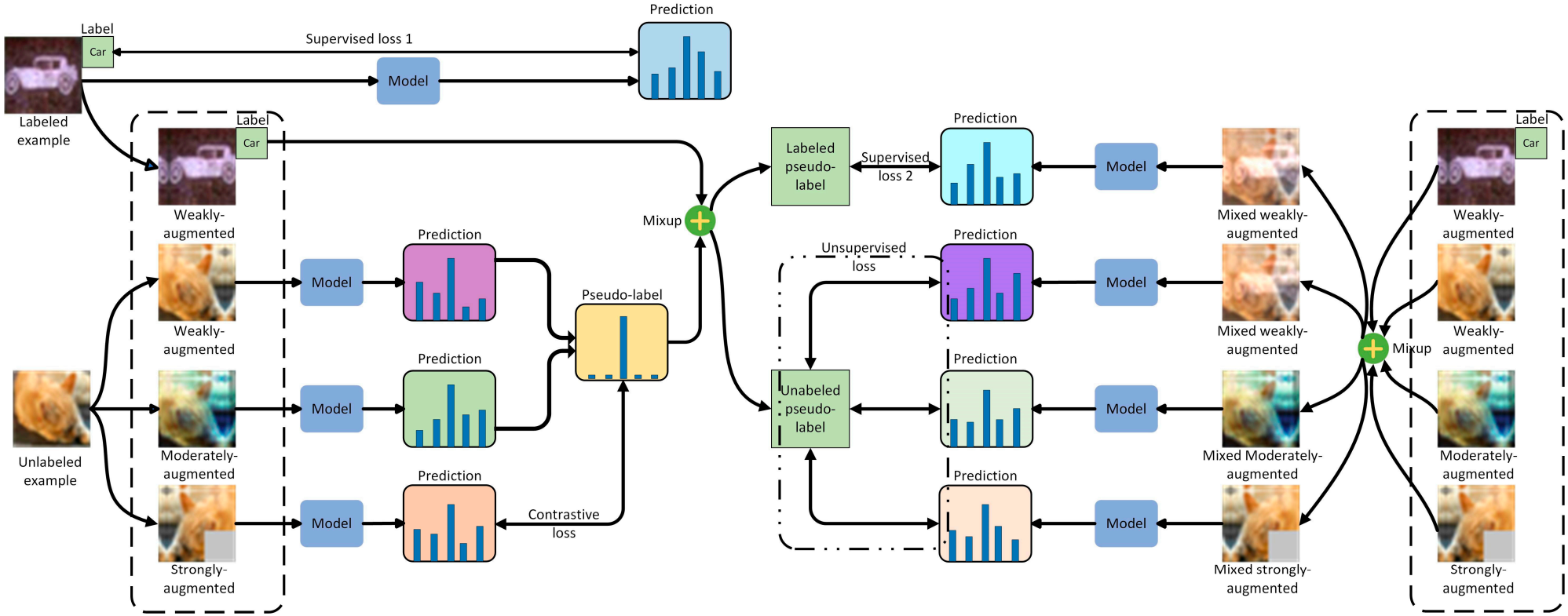


Figure 3. Diagram of MAC-SSL.

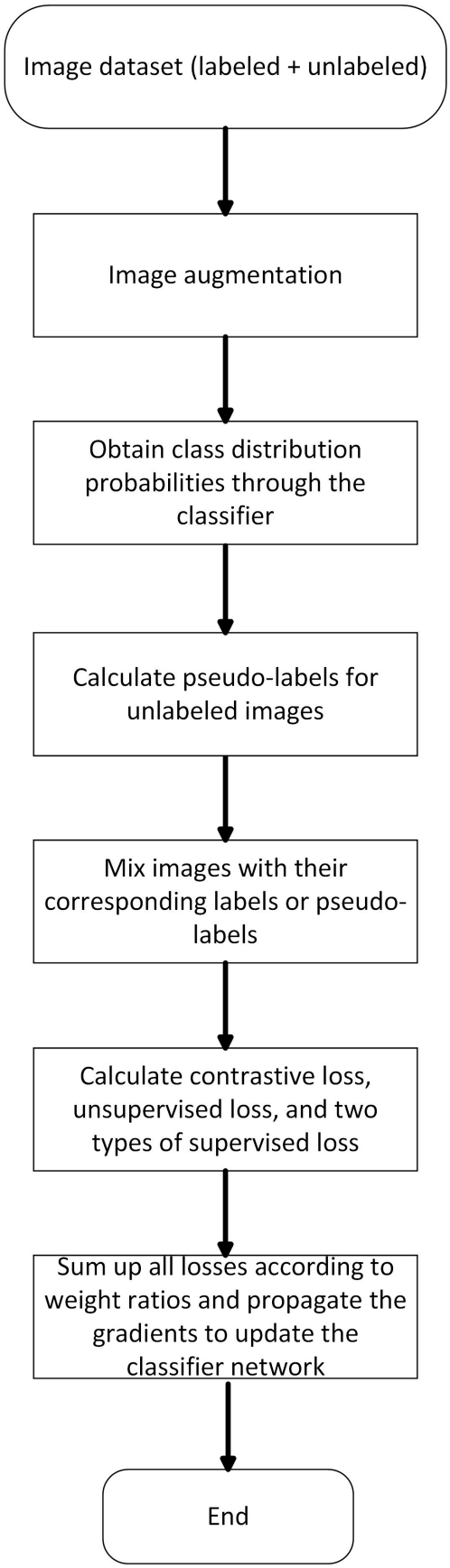


Figure 4. Flowchart of the MAC-SSL process.

The contrastive loss, $\mathcal{L}_c$, is established between $\mathcal{U}'_w$, $\mathcal{U}'_m$, and $\mathcal{U}'_s$. The object in this case is to maintain a certain level of strength in the optimization of the model on $\mathcal{U}'_s$. Therefore, a threshold, τ, is employed to p_b^u, and instead of converting p_b^u into pseudo-labels, the predicted distribution's state is directly utilized to constrain the model's output, $p(q|u'_{sb})$, which is also a predicted distribution, on $\mathcal{U}'_s$ using cross-entropy. $\mathcal{L}_c$ can be calculated as follows:

$$\mathcal{L}_c = \frac{1}{\mu B}\sum_{b=1}^{\mu B} f(\max(p_b^u) \geq \tau) H(p_b^u, p(q|u_{sb'})), \tag{15}$$

where $u'_{sb} \in \mathcal{U}'_s$ and $p(q|u'_{sb})$ are the model's predictions for u'_{sb}, H denotes the cross-entropy between the distribution p_b^u and $p(q|u'_{sb})$, and $f(\bullet)$ represents 1 if the inequality in the parentheses holds (and it is otherwise 0).

By combining the aforementioned loss functions, the loss function of MAC-SSL is defined as follows:

$$\mathcal{L} = \mathcal{L}_x + \mathcal{L}_{xm} + \lambda_u \mathcal{L}_u + \lambda_c \mathcal{L}_c, \tag{16}$$

where λ_u and λ_c are trade-off hyper-parameters that control the weights of the unsupervised loss $\mathcal{L}_u$ and the contrastive loss $\mathcal{L}_c$, respectively.

This design takes into account the different characteristics of the data by comprehensively utilizing different types of data and loss functions, enabling the model to learn the features of the data more comprehensively and, thus, improve its performance. Specifically, the benefits of such comprehensive utilization are reflected in the following aspects:

(1) Integration of information from different types of data: This model combines labeled and unlabeled data, applying different types of loss functions to both. $\mathcal{L}_x$ and $\mathcal{L}_{xm}$ utilize the true labels of labeled data, $\mathcal{L}_u$ utilizes pseudo-labels of unlabeled data, and $\mathcal{L}_c$ performs contrastive learning on unlabeled data. By comprehensively utilizing these different types of data and loss functions, the model can learn the features of the data more comprehensively, thereby improving its performance.

(2) Enhancing the model's generalization ability: Integrating different types of loss functions can improve the model's generalization ability. $\mathcal{L}_x$ and $\mathcal{L}_{xm}$ help the model learn accurate classification decisions on labeled data, while $\mathcal{L}_u$ and $\mathcal{L}_c$ can help the model utilize information from unlabeled data to enhance its generalization ability.

(3) Strengthening the model's understanding of the data: Different types of loss functions allow the model to understand the data from different perspectives. $\mathcal{L}_x$ and $\mathcal{L}_{xm}$ help the model understand the true labels of the data, while $\mathcal{L}_u$ and $\mathcal{L}_c$ enable the model to learn the distribution and features of the data from unlabeled data, thereby improving the model's robustness.

The full MAC-SSL algorithm is presented in Algorithm 2.

3. Results

MAC-SSL was evaluated on several benchmarks for SSL image classification (see Section 3.2). Furthermore, ablation experiments were conducted on each of MAC-SSL's components to analyze their individual contributions (see Section 3.3).

3.1. Implementation Details

1. Datasets and metrics: MAC-SSL was experimentally validated on the CIFAR-10 [42], CIFAR-100 [42], and SVHN [43] datasets, as shown in Figure 5.
    - CIFAR-10: The CIFAR-10 dataset is a widely used benchmark in the field of computer vision. It consists of 60,000 color images, each sized at 32 × 32 pixels, belonging to 10 different classes. These classes include common objects such as airplanes, automobiles, cats, birds, dogs, and more. The dataset is divided into 50,000 training images and 10,000 test images, with an equal distribution of images across the classes.

- CIFAR-100: Similar to CIFAR-10, CIFAR-100 is another dataset used for image classification tasks. It contains 60,000 images, each also sized at 32 × 32 pixels, but it is divided into 100 fine-grained classes. Each class represents a specific object or concept, such as insects, food containers, trees, fish, and so on. The dataset is split into 50,000 training images and 10,000 test images, with a balanced distribution of images across the classes. Classifying CIFAR-100 is more difficult than CIFAR-10 due to its larger number of categories.
- SVHN (Street View House Numbers): The SVHN dataset is focused on digit recognition tasks. It consists of real-world images taken from Google Street View that contain house numbers. Each image in SVHN contains multiple digits (from zero to nine). The images are of varying sizes but are predominantly sized at 32 × 32 pixels. SVHN is divided into a training set with 73,257 images and a test set with 26,032 images.

2. Experiment setting: All experiments were implemented using PyTorch and conducted on an Ubuntu system server with four NVIDIA 3090 GPUs and 128 GB of memory, and they followed SSL evaluation methods. The experiments were conducted using the "Wide ResNet-28" model from [44]. Training for the CIFAR-10 [42] and SVHN [43] datasets continued for 300 epochs until convergence. A batch size of 64 was used with the Wide ResNet-28-2 model, which has 1.47 M parameters. Due to computational limitations, a batch size of up to 32 was used for the CIFAR-100 [42] dataset in MAC-SSL, while for the other methods, the batch size remained at 64. The Wide ResNet-28-8 model with 23.46 M parameters was utilized for the CIFAR-100 training. For the selection of hyperparameters, we employed a random search. The hyperparameter μ, controlling the sample ratio, was set to 5. The weight hyperparameter λ_u was set as $\frac{currentepoch}{totalepoch} \times 75$, and λ_c was set to 1. The learning rate was set to 0.01 for CIFAR-10, CIFAR-100, and SVHN. The threshold τ was set to 0.95. The training employed the SGD optimizer with cosine weight decay. Exponential moving average (EMA) with a decay rate of 0.999 was utilized for evaluating the models. Of note, for SVHN, we applied strong augmentation using RandAugment [40] on top of moderate augmentation, referred to as MAC-SSL-II. This differed from CIFAR-10/100. Each epoch involved 1024 steps of training, and checkpoints were saved at each epoch. The average accuracy of the last 20 checkpoints was recorded. This approach simplified the analysis process.

(**a**) (**b**) (**c**)

Figure 5. (**a**) CIFAR-10 dataset. (**b**) CIFAR-100 dataset. (**c**) SVHN dataset.

For the SOTA algorithms used in the comparative experiments, we reproduced results by obtaining their source code and maintaining the original settings mentioned in the code. For some datasets that were not included in their original code, we added them ourselves and used the hyperparameters specified in the respective papers. The source code for the

comparative algorithms can be downloaded from the authors' or reproducers' homepages, except for Mean-Teacher [32] (reproduced by ourselves). The codes for CoMatch[1] [34], MixMatch[2] [33], ICT[3] [45], VAT[3] [46], Temporal-ensembling[3] [30], and Pimodel[3] [30] are available on their respective authors' or reproducers' homepages.

3.2. Comparison Algorithms

(1) To demonstrate the superiority of MAC-SSL, performance comparison experiments were conducted with the following seven state-of-the-art semi-supervised algorithms:
(2) CoMatch [34] combines pseudo-based, contrast-loss-based, and graph-based models to improve model performance with limited labeled data. It jointly learns class probabilities and low-dimensional embeddings, enhancing the quality of pseudo-labels by imposing a smoothness constraint on the class probabilities.
(3) MixMatch [33] optimizes both supervised and unsupervised losses. It utilizes cross-entropy for supervised losses and mean square errors (MSEs) between predictions and generated pseudo-labels for unsupervised losses. MixMatch constructs pseudo-labels through data augmentation and improves their quality using the Sharpen function. MixUP [38] interpolation is also employed to create virtual samples.
(4) Mean-Teacher [32] employs a student–teacher approach for SSL. The teacher model is based on the average weights of a student model in each update step. Mean-Teacher utilizes MSE loss as the consistency loss between two predictions and updates the model using exponential moving average (EMA) to control the update speed.
(5) ICT [45] extends MixUP by interpolating unlabeled data, generating diverse mixed samples. It enforces consistency across different interpolation ratios using regularization. ICT trains a model by constraining predictions of mixed data to align with mixed predictions of original data. It effectively utilizes unlabeled data, particularly in scenarios with limited labeled data, resulting in improved generalization capabilities.
(6) VAT [46] replaces data augmentation with adversarial transformations. It perturbs input data through adversarial transformations, leading to lower classification errors.
(7) Temporal-ensembling [30] is a method based on temporal ensembling that improves model consistency and robustness by using an exponential moving average of historical prediction results during training. It trains the model by minimizing the consistency loss between the predictions of unlabeled data and the true labels of labeled data.
(8) Pimodel [30] is a method based on data augmentation and consistency regularization. It generates virtual samples using data augmentation and trains the model by applying consistency constraints between labeled and unlabeled data.

3.3. Performance Comparison

1. CIFAR-10: For CIFAR-10, performance comparison experiments were conducted with six baselines, including MixMatch [33], Mean-Teacher [32], ICT [45], VAT [46], Temporal-ensembling [30], and Pimodel [30]. The accuracy of these methods was evaluated with a varying number of labeled samples from 250 to 4000 (as is standard practice). The result can be seen in Figure 6. It can be observed that MAC-SSL was significantly superior to all other methods, especially when labeled samples were scarce, such as with 250 labels and 500 labels. MAC-SSL outperformed the second-best method, MixMatch, by 6.82% and 7.02%, respectively. These results highlight MAC-SSL's ability to effectively utilize information from unlabeled data, thereby delivering a strong performance even in a scenario with limited labeled samples.

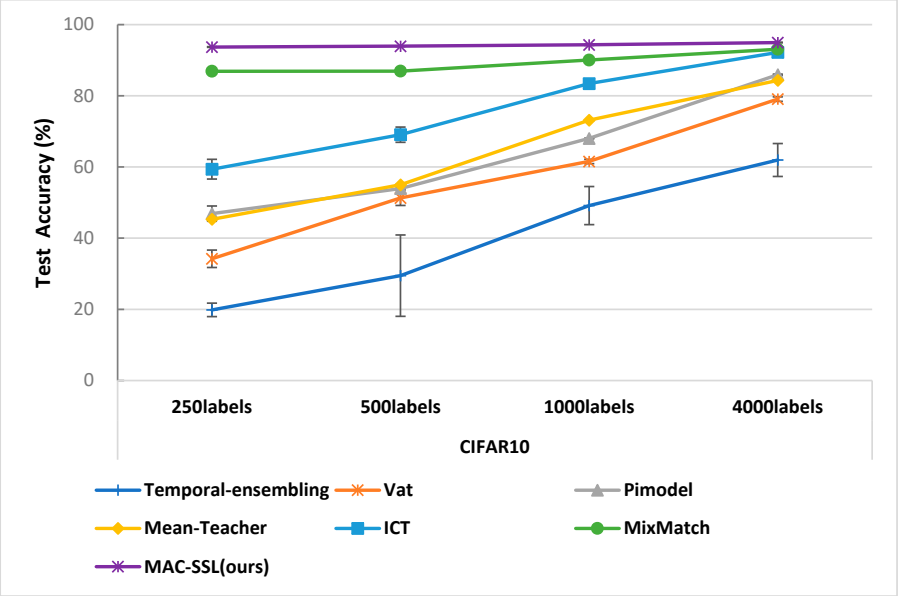


Figure 6. Accuracy (%) of CIFAR-10. Exact numbers are provided in Table A1 (Appendix A).

2. CIFAR-100: To further demonstrate the effectiveness of MAC-SSL, we conducted comparative experiments on CIFAR-100. The baselines for comparison were the same as those used for CIFAR-10. The CIFAR-100 evaluation involved a varying number of labeled samples ranging from 400 to 2500. The results are presented in Figure 7. Upon observation, it can be seen that MAC-SSL achieved the best performance. When the number of labeled samples was 400, which means only 4 labeled samples per class, MAC-SSL outperformed the second-best method, MixMatch, by 16.58%. It is also noteworthy that as the number of labeled samples decreased, MAC-SSL brought even greater improvements, further validating the ability of the proposed method to effectively utilize information from unlabeled samples.

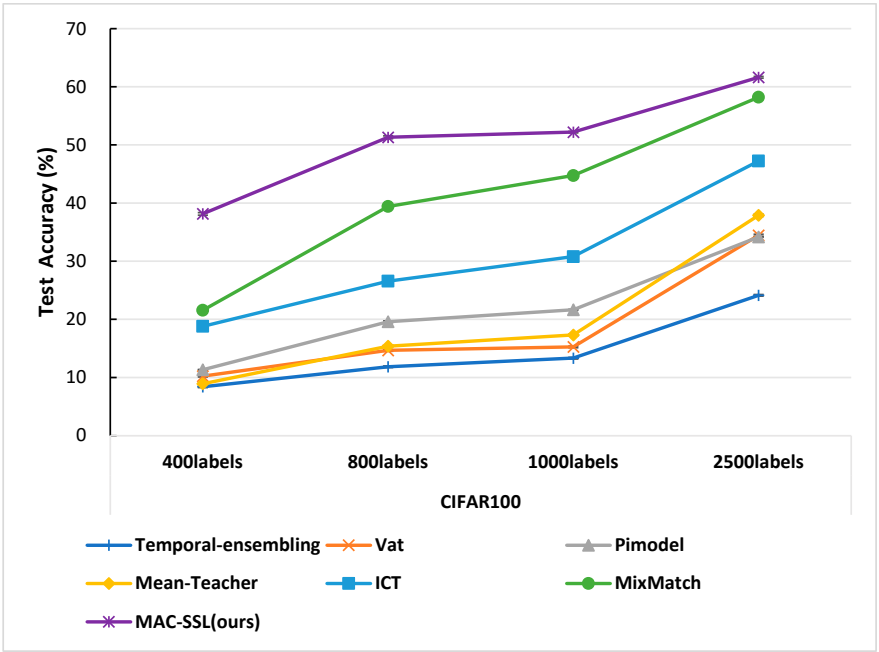


Figure 7. Accuracy (%) of CIFAR-100. Exact numbers are provided in Table A2 (Appendix A).

3. SVHN: We conducted comparative experiments on the SVHN dataset and, in addition to CIFAR-10 and CIFAR-100, included CoMatch as one of the baselines. The accuracy of these methods was evaluated with varying numbers of labeled samples ranging from 250 to 4000. The experimental results are presented in Figure 8. As observed earlier, MAC-SSL achieved the best results and demonstrated greater improvements when the number of labeled samples was reduced. This once again confirmed the effectiveness of MAC-SSL and its ability to fully utilize unlabeled samples.

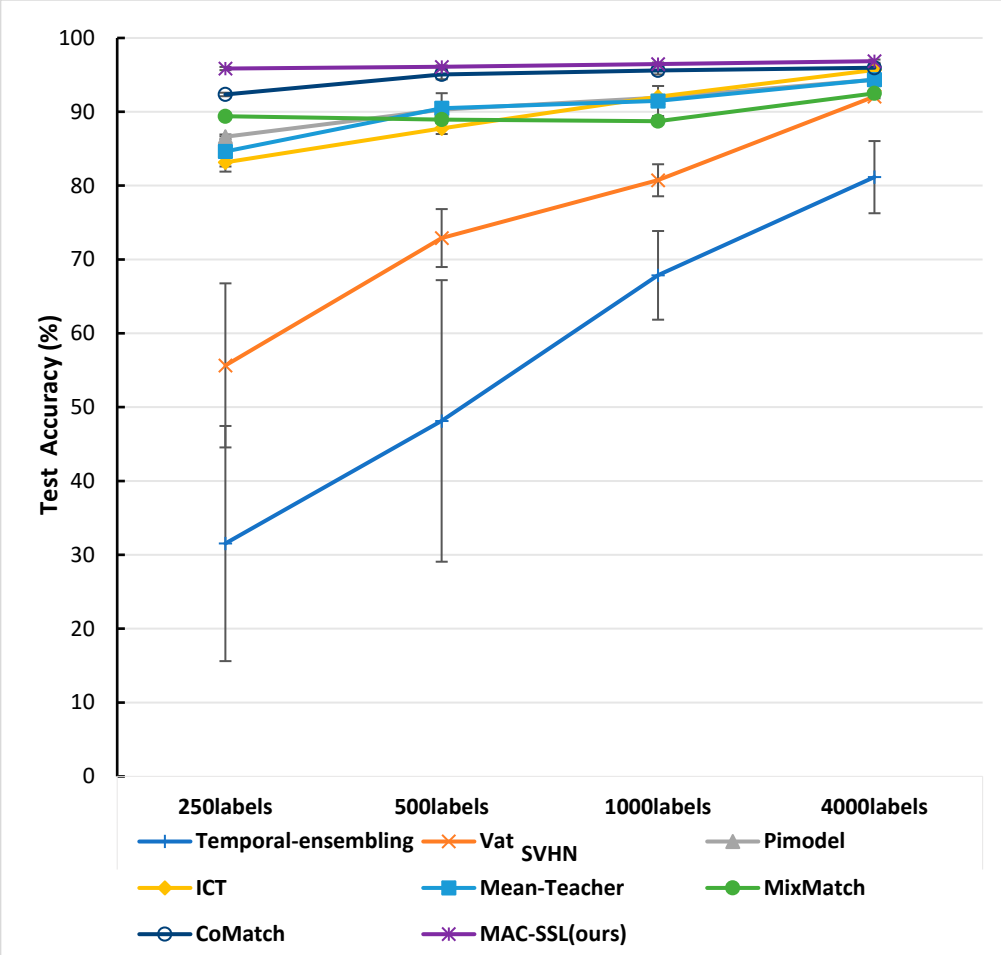
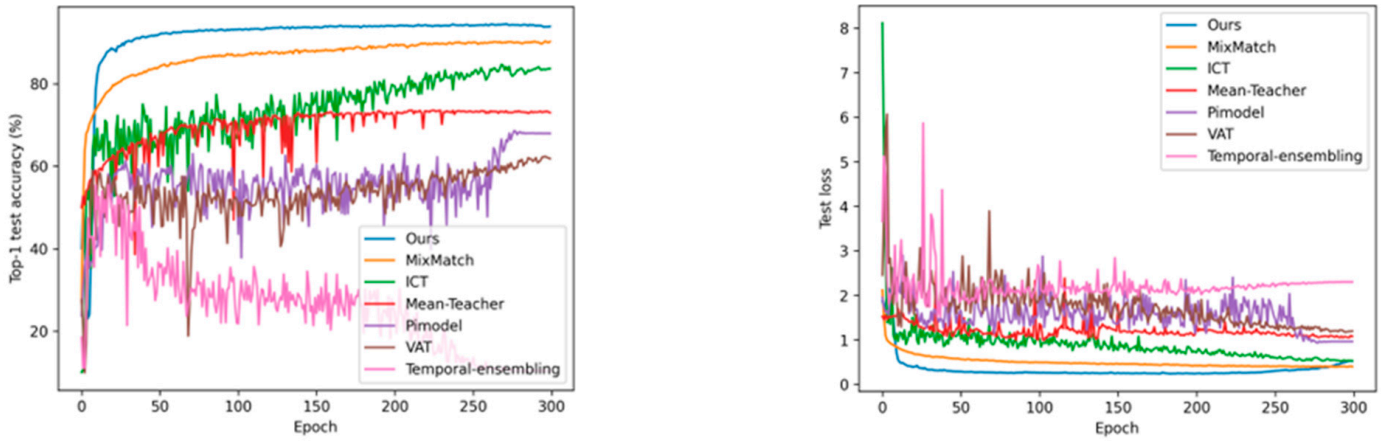


Figure 8. Accuracy (%) of SVHN. Exact numbers are provided in Table A3 (Appendix A).

Through the conducted experiments, the proposed method demonstrated superior performance compared to all other existing methods, particularly in scenarios with limited availability of labeled samples. The key advantage of our approach was its ability to effectively leverage the valuable information contained within unlabeled samples, which is often overlooked by other methods' data augmentation techniques. MAC-SSL stood out by generating reliable positive sample pairs through three distinct augmentation techniques, thereby enhancing the regularization benefits. Furthermore, by incorporating MixUP with augmented images and their corresponding labels or pseudo-labels, our model exhibited enhanced generalization capabilities and achieved efficient convergence.

Notably, MAC-SSL exhibited exceptional classification performances across multiple benchmark datasets, including CIFAR-10, CIFAR-100, and SVHN. These results highlight the versatility and robustness of our approach, especially in scenarios where labeled samples are scarce. The promising outcomes achieved by MAC-SSL validate its potential as a valuable contribution to the field, addressing the challenges associated with limited labeled data and demonstrating its efficacy in improving classification performance.

We also present in Figure 9 the test accuracy and test loss during the training process on the CIFAR-10 dataset with 1000 labeled samples using the different methods. It can be observed that MAC-SSL achieved rapid convergence, attaining the highest accuracy and nearly the lowest loss (with only a slight increase in loss towards the end). This further demonstrated the effectiveness and stability of MAC-SSL.



(**a**) (**b**)

Figure 9. Plots of the different methods as training progress on CIFAR-10 with 1000 labeled samples. (**a**) Accuracy on the test data. (**b**) Loss on the test data.

3.4. Ablation Experiment

Additional ablation experiments were conducted to examine the roles of the different components in MAC-SSL and evaluate the impacts of certain parameters. Specifically, we measured the effects of the following:

(1) Investigating the effects of different strong augmentation strategies: directly using RandAugment (MAC-SSL) and using RandAugment after applying moderate augmentation (MAC-SSL-II)
(2) Using different sample ratio hyperparameter μ values ranging from 1 to 9
(3) Removing temperature sharpening (i.e., setting $T = 1$)
(4) Performing MixUP between labeled examples only, unlabeled examples only, and without mixing across labeled and unlabeled examples
(5) Using the mean class distribution over two augmentations (i.e., weak and moderate augmentations) or using the class distribution for a single augmentation (i.e., only weak augmentation)
(6) Employing weak augmentation, moderate augmentation, and strong augmentation, as well as the scenario where only weak augmentation and strong augmentation were utilized (with the class distribution being used only for weak augmentation)

The ablation experiments were conducted on the CIFAR-10 and SVHN datasets, and the results are shown in Tables 1–3. It was observed that each component contributed to the performance of MAC-SSL, and incorporating MixUP solely on labeled samples resulted in the largest performance loss, even surpassing the impact caused by not using MixUP at all. This indicated that unlabeled samples contributed significantly to the training process, as the information contained within a large volume of unlabeled samples exceeded that of a small number of labeled samples. MAC-SSL was able to fully utilize the information embedded within the unlabeled samples, resulting in remarkable performance.

Table 1. Ablation study results (a). All values indicate accuracy (%).

Method	CIFAR-10 (μ = 5)			
	40 Labels	**500 Labels**	**1000 Labels**	**4000 Labels**
MAC-SSL (base for CIFAR10)	87.36 ± 0.09	**93.93 ± 0.1**	**94.32 ± 0.18**	**94.96 ± 0.09**
MAC-SSL-II (base for SVHN)	**90.22 ± 0.14**	93.29 ± 0.07	93.97 ± 0.08	94.91 ± 0.07
MAC-SSL/MAC-SSL-II without temperature sharpening (T = 1)	–	–	93.31 ± 0.08	–
MAC-SSL/MAC-SSL-II without MixUP	–	–	91.88 ± 0.11	–
MAC-SSL/MAC-SSL-II with MixUP on labeled only	–	–	90.07 ± 0.14	–
MAC-SSL/MAC-SSL-II with MixUP on unlabeled only	–	–	93.97 ± 0.35	–
MAC-SSL/MAC-SSL-II with MixUP on separate labeled and unlabeled	–	–	92.36 ± 0.42	–
MAC-SSL/MAC-SSL-II without distribution averaging	–	–	91.6 ± 0.18	–
MAC-SSL/MAC-SSL-II without moderate augmentation	–	–	94.18 ± 0.24	–

Table 2. Ablation study results (b). All values indicate accuracy (%). For SVHN, MAC-SSL-II was the base.

Method	SVHN (μ = 5)	
	250 Labels	**1000 Labels**
MAC-SSL (base for CIFAR10)	95.35 ± 0.1	95.65 ± 0.08
MAC-SSL-II (base for SVHN)	**95.87 ± 0.06**	**96.46 ± 0.04**
MAC-SSL/MAC-SSL-II without temperature sharpening (T = 1)	–	95.95 ± 0.05
MAC-SSL/MAC-SSL-II without MixUP	–	94.83 ± 0.12
MAC-SSL/MAC-SSL-II with MixUP on labeled only	–	94.02 ± 0.13
MAC-SSL/MAC-SSL-II with MixUP on unlabeled only	–	96.17 ± 0.06
MAC-SSL/MAC-SSL-II with MixUP on separate labeled and unlabeled	–	96.06 ± 0.04
MAC-SSL/MAC-SSL-II without distribution averaging	–	96.44 ± 0.04
MAC-SSL/MAC-SSL-II without moderate augmentation	–	95.50 ± 0.06

Table 3. Ablation study results (c). All values indicate accuracy (%).

Method	CIFAR-10 (1000 Labels)				
	μ = 1	**μ = 3**	**μ = 5**	**μ = 7**	**μ = 9**
MAC-SSL	**89.47 ± 0.14**	**93.18 ± 0.08**	**94.32 ± 0.18**	**94.55 ± 0.08**	**96.46 ± 0.04**
MAC-SSL-II	88.49 ± 0.18	92.96 ± 0.17	93.96 ± 0.08	94.5 ± 0.09	94.85 ± 0.11

Applying RandAugment in addition to moderate augmentation as the strong augmentation strategy proved to be more effective on the SVHN dataset. MAC-SSL also yielded superior results with a small number of labeled samples and an increased number of unlabeled samples (an increased μ). Notably, it outperformed the conventional strong augmentation strategy on the CIFAR-10 dataset with 40 labeled samples, as well as on the CIFAR-10 dataset with 1000 labeled samples when $\mu = 9$.

4. Discussion

The paper introduces MAC-SSL, which proposes a novel approach to leverage the valuable information within unlabeled samples in semi-supervised learning tasks. Unlike traditional methods that rely on pseudo-labels or soft labels to enforce consistency between strongly and weakly augmented unlabeled images, MAC-SSL incorporates a moderate augmentation step. This step combines the model outputs of moderately augmented unlabeled images with those of weakly augmented unlabeled images to generate pseudo-labels. Subsequently, the model outputs of strongly augmented images are enforced to be consistent with the pseudo-labels. Additionally, inspired by MixMatch, MixUP is adopted

to combine the outputs of unlabeled images with those of labeled images, augmenting the data and ensuring consistency between the model outputs of the re-augmented images and the newly generated pseudo-labels derived from mixed labels or pseudo-labels.

The experimental results demonstrate the superior performance of MAC-SSL compared to existing methods, particularly in scenarios with limited labeled samples. MAC-SSL effectively leverages the valuable information within unlabeled samples, which other methods often overlook in their data augmentation techniques. MAC-SSL generates reliable positive sample pairs through three distinct augmentation techniques, enhancing the regularization benefits. Moreover, the integration of MixUP with augmented images and their corresponding labels or pseudo-labels improves the generalization capabilities of the model and facilitates efficient convergence.

Evaluation on the CIFAR-10, CIFAR-100, and SVHN benchmark datasets consistently showed that MAC-SSL outperformed the other methods, especially with limited labeled samples. For CIFAR-10, MAC-SSL achieved significant accuracy improvements over the second-best method, MixMatch, with increases of 6.82% and 7.02% when only 250 and 500 labeled samples were available, respectively. Similarly, MAC-SSL surpassed MixMatch by 16.58% on CIFAR-100 even with as few as 400 labeled samples. The experiments on the SVHN dataset further validated the effectiveness of MAC-SSL in fully leveraging unlabeled samples for a superior performance.

The ablation studies provided insights into the roles of different components in MAC-SSL and the impacts of various parameters. Utilizing unlabeled samples through MixUP was found to be crucial, as removing MixUP solely on unlabeled samples resulted in an even greater performance loss compared to not using MixUP. Furthermore, MAC-SSL demonstrated superior results with a small number of labeled samples and an increased number of unlabeled samples, highlighting the importance of leveraging a large volume of unlabeled samples. The choice of a strong augmentation strategy, the sample ratio hyperparameter, the temperature sharpening, and the combination of weak, moderate, and strong augmentations also impacted the performance of MAC-SSL, showcasing the effectiveness of these design choices.

While the increased augmentation consumes more computational resources, the performance improvements achieved by MAC-SSL justify the additional cost. Future research can explore strategies to optimize the computational efficiency of MAC-SSL without compromising its effectiveness, such as excluding backpropagation on weakly augmented unlabeled samples. Additionally, the current trend of using large-scale models suggests that employing pre-trained models will lead to further breakthroughs in the future. Finally, larger and more complex datasets such as ImageNet [47], as well as specialized and widely used datasets for semi-supervised learning such as medical image datasets, are also within our scope of consideration. In the future, we plan to conduct more extensive experiments on these datasets.

In conclusion, the experiments and ablation studies demonstrated the superiority of MAC-SSL in semi-supervised learning scenarios, particularly with limited labeled samples. MAC-SSL effectively leverages the valuable information within unlabeled samples, resulting in improved classification performances across multiple benchmark datasets. The integration of moderate augmentation, collaboration between weak and moderate augmented unlabeled samples, and the use of MixUP enhance the regularization benefits and generalization capabilities of MAC-SSL. MAC-SSL makes a valuable contribution to the field by addressing the challenges associated with limited labeled data, demonstrating its efficacy in improving classification performance. Future research can explore extensions and modifications to further enhance MAC-SSL's performance or investigate its applicability in other domains and datasets.

5. Conclusions

Herein, we proposed MAC-SSL, a semi-supervised learning method that leverages contrastive learning with multiple augmentation strategies. Through extensive comparative experiments in the field of semi-supervised learning, MAC-SSL demonstrated remarkable performance improvements compared to other methods, showcasing its effectiveness and generality. Ablation experiments further emphasized the significance of each component in MAC-SSL and uncovered its potential for further enhancements, such as achieving a better performance with a larger μ. Moving forward, we are interested in exploring the effectiveness of MAC-SSL in different domains. Additionally, we aim to streamline the algorithm while maintaining its performance and investigate methods to ensure its effectiveness in scenarios with fewer labeled samples.

In future work, the application of MAC-SSL to different domains and its effectiveness in those domains will be explored. Additionally, efforts will be made to simplify the algorithm while maintaining its performance to improve usability. Furthermore, research will be conducted to ensure the effectiveness of the method in scenarios with even fewer labeled samples. Lastly, in an era of large models, the utilization of pre-training models will also be further explored.

Author Contributions: Conceptualization, J.W.; methodology, J.W. and J.Y.; software, J.Y. and D.P.; validation, J.W., J.Y. and J.H.; formal analysis, J.H.; investigation, J.Y. and D.P.; resources, D.P.; data curation, J.W.; writing—original draft preparation, J.W. and J.Y.; writing—review and editing, J.Y. and D.P.; supervision, J.Y. and D.P.; project administration, J.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Science and Technology on Information System Engineering Laboratory under grant 05202206, by the Fundamental Research Funds for the provincial Universities of Zhejiang under grant GK229909299001-024, by the Key Laboratory of Avionics System Integrated Technology, and in part by the National Natural Science Foundation of China under grant U22A2047.

Data Availability Statement: The CIFAR-10, CIFAR-100, and SVHN datasets used in this paper are all public datasets.

Acknowledgments: All authors would like to thank the editors and reviewers for their detailed comments and suggestions.

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

Appendix A

Table A1. Accuracy (%) of CIFAR-10. The bold values indicate the best and the runner-up results.

Method	CIFAR-10			
	250 Labels	**500 Labels**	**1000 Labels**	**4000 Labels**
Temporal-ensembling [30]	19.86 ± 1.9	29.48 ± 11.44	49.16 ± 5.34	61.97 ± 4.61
Vat [46]	34.22 ± 2.46	51.3 ± 2.11	61.56 ± 0.61	79.08 ± 0.62
Pimodel [30]	46.9 ± 2.15	53.93 ± 0.16	68.01 ± 0.08	85.92 ± 0.17
Mean-Teacher [32]	45.3 ± 0.1	54.98 ± 0.12	73.14 ± 0.11	84.34 ± 0.08
ICT [45]	59.38 ± 2.78	69.07 ± 2.15	83.43 ± 0.4	92.2 ± 0.3
MixMatch [33]	**86.88 ± 0.34**	**86.91 ± 0.17**	**90.03 ± 0.16**	**93.08 ± 0.08**
MAC-SSL (ours)	**93.7 ± 0.1**	**93.93 ± 0.1**	**94.32 ± 0.18**	**94.96 ± 0.09**

Table A2. Accuracy (%) of CIFAR-100. The bold values indicate the best and the runner-up results.

Method	CIFAR-100			
	400 Labels	**800 Labels**	**1000 Labels**	**2500 Labels**
Temporal-ensembling [30]	8.42 ± 0.14	11.85 ± 0.14	13.33 ± 0.14	24.15 ± 0.14
Vat [46]	10.23 ± 0.13	14.67 ± 0.14	15.27 ± 0.15	34.46 ± 0.19
Pimodel [30]	11.35 ± 0.04	19.59 ± 0.15	21.66 ± 0.07	34.19 ± 0.05
Mean-Teacher [32]	8.93 ± 0.09	15.38 ± 0.08	17.31 ± 0.13	37.9 ± 0.18
ICT [45]	18.81 ± 0.22	26.57 ± 0.16	30.79 ± 0.73	56.53 ± 0.42
MixMatch [33]	**21.57 ± 0.2**	**39.42 ± 0.18**	**44.74 ± 0.14**	**58.21 ± 0.19**
MAC-SSL (ours)	**38.15 ± 0.22**	**51.32 ± 0.25**	**52.20 ± 0.13**	**61.64 ± 0.16**

Table A3. Accuracy (%) of SVHN. The bold values indicate the best and the runner-up results.

Method	SVHN			
	250 Labels	**500 Labels**	**1000 Labels**	**4000 Labels**
Temporal-ensembling [30]	31.54 ± 15.92	48.14 ± 19.06	67.86 ± 6.01	81.16 ± 4.89
Vat [46]	55.66 ± 11.11	72.9 ± 3.92	80.74 ± 2.17	92.07 ± 0.25
Pimodel [30]	86.62 ± 0.31	90.29 ± 0.05	91.9 ± 0.08	94.34 ± 0.11
ICT [45]	83.17 ± 1.27	87.75 ± 0.75	92.03 ± 0.22	95.66 ± 0.16
Mean-Teacher [32]	84.64 ± 0.08	90.48 ± 0.07	91.47 ± 0.05	94.36 ± 0.04
MixMatch [33]	89.39 ± 0.34	88.94 ± 0.3	88.73 ± 0.38	92.51 ± 0.08
CoMatch [34]	**92.35 ± 0.24**	**95.06 ± 0.66**	**95.6 ± 0.54**	**95.96 ± 0.42**
MAC-SSL (ours)	**95.87 ± 0.06**	**96.08 ± 0.07**	**96.46 ± 0.04**	**96.85 ± 0.04**

References

1. Huang, X.; Song, Z.; Ji, C.; Zhang, Y.; Yang, L. Research on a Classification Method for Strip Steel Surface Defects Based on Knowledge Distillation and a Self-Adaptive Residual Shrinkage Network. *Algorithms* **2023**, *16*, 516. [CrossRef]
2. He, R.; Han, Z.; Lu, X.; Yin, Y. Safe-student for safe deep semi-supervised learning with unseen-class unlabeled data. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 19–24 June 2022; pp. 14585–14594.
3. Zhang, L.; Xiong, N.; Pan, X.; Yue, X.; Wu, P.; Guo, C. Improved Object Detection Method Utilizing YOLOv7-Tiny for Unmanned Aerial Vehicle Photographic Imagery. *Algorithms* **2023**, *16*, 520. [CrossRef]
4. Wu, H.; Ma, X.; Liu, S. Designing multi-task convolutional variational autoencoder for radio tomographic imaging. *IEEE Trans. Circuits Syst. II Express Briefs* **2021**, *69*, 219–223. [CrossRef]
5. Pacella, M.; Mangini, M.; Papadia, G. Utilizing Mixture Regression Models for Clustering Time-Series Energy Consumption of a Plastic Injection Molding Process. *Algorithms* **2023**, *16*, 524. [CrossRef]
6. Wang, S.; Liu, X.; Liu, L.; Tu, W.; Zhu, X.; Liu, J.; Zhou, S.; Zhu, E. Highly-efficient incomplete large-scale multi-view clustering with consensus bipartite graph. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 19–24 June 2022; pp. 9776–9785.
7. Zhu, F.; Zhao, J.; Cai, Z. A Contrastive Learning Method for the Visual Representation of 3D Point Clouds. *Algorithms* **2022**, *15*, 89. [CrossRef]
8. Ali, O.; Ali, H.; Shah, S.A.A.; Shahzas, A. Implementation of a modified U-Net for medical image segmentation on edge devices. *IEEE Trans. Circuits Syst. II Express Briefs* **2022**, *69*, 4593–4597. [CrossRef]
9. Hu, X.; Zeng, Y.; Xu, X.; Zhou, S.; Liu, L. Robust semi-supervised classification based on data augmented online ELMs with deep features. *Knowl. Based Syst.* **2021**, *229*, 107307. [CrossRef]
10. Lindstrom, M.R.; Ding, X.; Liu, F.; Somayajula, A.; Needell, D. Continuous Semi-Supervised Nonnegative Matrix Factorization. *Algorithms* **2023**, *16*, 187. [CrossRef]
11. Yang, J.; Cao, J.; Xue, A. Robust maximum mixture correntropy criterion-based semi-supervised ELM with variable center. *IEEE Trans. Circuits Syst. II Express Briefs* **2020**, *67*, 3572–3576. [CrossRef]
12. Saito, K.; Kim, D.; Sclaroff, S.; Darrell, T.; Saenko, K. Semi-supervised domain adaptation via minimax entropy. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 8050–8058.
13. Kipf, T.N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv* **2016**, arXiv:1609.02907.
14. Hu, Z.; Yang, Z.; Hu, X.; Nevatia, R. Simple: Similar pseudo label exploitation for semi-supervised classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Online, 19–25 June 2021; pp. 15099–15108.

15. Cai, J.; Hao, J.; Yang, H.; Zhao, X.; Yang, Y. A review on semi-supervised clustering. *Inf. Sci.* **2023**, *632*, 164–200. [CrossRef]
16. Chen, X.; Yuan, Y.; Zeng, G.; Wang, J. Semi-supervised semantic segmentation with cross pseudo supervision. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Online, 19–25 June 2021; pp. 2613–2622.
17. Xu, M.; Zhang, Z.; Hu, H.; Wang, J.; Wang, L.; Wei, F.; Bai, X.; Liu, Z. End-to-end semi-supervised object detection with soft teacher. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 3060–3069.
18. Kostopoulos, G.; Karlos, S.; Kotsiantis, S.; Ragos, O. Semi-supervised regression: A recent review. *J. Intell. Fuzzy Syst.* **2018**, *35*, 1483–1500. [CrossRef]
19. Van Engelen, J.E.; Hoos, H.H. A survey on semi-supervised learning. *Mach. Learn.* **2020**, *109*, 373–440. [CrossRef]
20. 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.
21. 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.
22. Xie, Q.; Luong, M.T.; Hovy, E.; Le, Q.V. Self-training with noisy student improves imagenet classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 10687–10698.
23. Rosenberg, C.; Hebert, M.; Schneiderman, H. Semi-supervised self-training of object detection models. In Proceedings of the 2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05), Breckenridge, CO, USA, 5–7 January 2005; Volume 1, pp. 29–36.
24. Zhai, X.; Oliver, A.; Kolesnikov, A.; Beyer, L. S4l: Self-supervised semi-supervised learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1476–1485.
25. Li, X.; Sun, Q.; Liu, Y.; Zhou, Q.; Zheng, S.; Chua, T.S.; Schiele, B. Learning to self-train for semi-supervised few-shot classification. In Proceedings of the Advances in Neural Information Processing Systems: 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, BC, Canada, 8 December 2019; p. 32.
26. Tanha, J.; Van Someren, M.; Afsarmanesh, H. Semi-supervised self-training for decision tree classifiers. *Int. J. Mach. Learn. Cybern.* **2017**, *8*, 355–370. [CrossRef]
27. Moskalenko, V.; Kharchenko, V.; Moskalenko, A.; Petrov, S. Model and Training Method of the Resilient Image Classifier Considering Faults, Concept Drift, and Adversarial Attacks. *Algorithms* **2022**, *15*, 384. [CrossRef]
28. Zhang, H.; Zhang, Z.; Odena, A.; Lee, H. Consistency regularization for generative adversarial networks. *arXiv* **2019**, arXiv:1910.12027.
29. Bachman, P.; Alsharif, O.; Precup, D. Learning with pseudo-ensembles. In Proceedings of the Advances in Neural Information Processing Systems: Annual Conference on Neural Information Processing Systems 2014, Montreal, QC, Canada, 8–13 December 2014; p. 27.
30. Laine, S.; Aila, T. Temporal ensembling for semi-supervised learning. *arXiv* **2016**, arXiv:1610.02242.
31. Zahedi, E.; Saraee, M.; Masoumi, F.S.; Yazdinejad, M. Regularized Contrastive Masked Autoencoder Model for Machinery Anomaly Detection Using Diffusion-Based Data Augmentation. *Algorithms* **2023**, *16*, 431. [CrossRef]
32. Tarvainen, A.; Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017; p. 30.
33. 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: 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, BC, Canada, 8–14 December 2019; p. 32.
34. Li, J.; Xiong, C.; Hoi, S.C. Comatch: Semi-supervised learning with contrastive graph regularization. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 9475–9484.
35. Antoniou, A.; Storkey, A.; Edwards, H. Data augmentation generative adversarial networks. *arXiv* **2017**, arXiv:1711.04340.
36. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 1–48. [CrossRef]
37. Zhong, Z.; Zheng, L.; Kang, G.; Li, S.; Yang, Y. Random erasing data augmentation. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), New York, NY, USA, 7–12 February 2020; Volume 34, pp. 13001–13008.
38. Zhang, H.; Cisse, M.; Dauphin, Y.N.; Lopez-Paz, D. mixup: Beyond empirical risk minimization. *arXiv* **2017**, arXiv:1710.09412.
39. Hendrycks, D.; Mu, N.; Cubuk, E.D.; Zoph, B.; Gilmer, J.; Lakshminarayanan, B. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv* **2019**, arXiv:1912.02781.
40. 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 (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 702–703.
41. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
42. Krizhevsky, A.; Hinton, G. *Learning Multiple Layers of Features from Tiny Images*; Technical Report; University of Toronto: Toronto, ON, Canada, 2009; pp. 1–58.
43. Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; Ng, A.Y. Reading digits in natural images with unsupervised feature learning. In Proceedings of the NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011, Granada, Spain, 12–17 December 2011.

44. Zagoruyko, S.; Komodakis, N. Wide residual networks. *arXiv* **2016**, arXiv:1605.07146.
45. Verma, V.; Kawaguchi, K.; Lamb, A.; Kannala, J.; Solin, A.; Bengio, Y.; Lopez-Paz, D. Interpolation consistency training for semi-supervised learning. *Neural Netw.* **2022**, *145*, 90–106. [CrossRef]
46. Miyato, T.; Maeda, S.I.; 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]
47. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, 20–25 June 2009; pp. 248–255.