Semi-ViM: Bidirectional State Space Model for Mitigating Label Imbalance in Semi-Supervised Learning

Hongyang He*
University of Warwick

Hongyang Xie
University of Warwick

Haochen You
Columbia University

Victor Sanchez
University of Warwick

Abstract

Semi-supervised learning (SSL) is often hindered by learning biases when imbalanced datasets are used for training, which limits its effectiveness in real-world applications. In this paper, we propose Semi-ViM, a novel SSL framework based on Vision Mamba, a bidirectional state space model (SSM) that serves as a superior alternative to Transformer-based architectures for visual representation learning. Semi-ViM effectively deals with imbalanced datasets and improves model stability through two key innovations: LyapEMA, a stability-aware parameter update mechanism inspired by Lyapunov theory, and SSMixup, a novel mixup strategy applied at the hidden state level of bidirectional SSMs. Experimental results on ImageNet-1K and ImageNet-LT demonstrate that Semi-ViM significantly outperforms state-of-the-art SSL models, achieving 85.40% accuracy with only 10% of the labeled data, surpassing Transformer-based methods such as Semi-ViT.

1. Introduction

Semi-supervised learning (SSL) aims to mitigate the scarcity of large-scale labeled training data while reducing annotation costs [16]. In the context of classification, however, a challenge remains: Most state-of-the-art (SOTA) SSL models are trained under the assumption that access to a dataset with a balanced number of labeled samples per class is possible, which is often overly idealized and deviates significantly from real-world conditions [20]. In practice, label imbalance induces learning biases in SSL, ultimately compromising model performance [41].

Due to its outstanding performance, many studies have integrated the vision Transformer (ViT) into SSL methods [4, 43, 46]; e.g., Semi-ViT, adopts a pre-training and fine-tuning approach to achieve SOTA results [4]. However, Semi-ViT has a weaker inductive bias, requiring large-scale data for effective training [39]. Moroever, as shown by our experiments, Semi-ViT does not perform well on long-tailed imbalanced datasets, like ImageNet-LT, confirming its limitations in handling label imbalance. Although the performance of Semi-ViT can be improved by increasing the model size, e.g., Semi-ViT-Huge, this comes at the expense of a larger number of trainable parameters, around 632M [4]. Vision Mamba (ViM), which incorporates a bidirectional state space model (SSM), has modeling capabilities comparable to those of ViT, while achieving subquadratic-time computational and linear memory complexities [47].

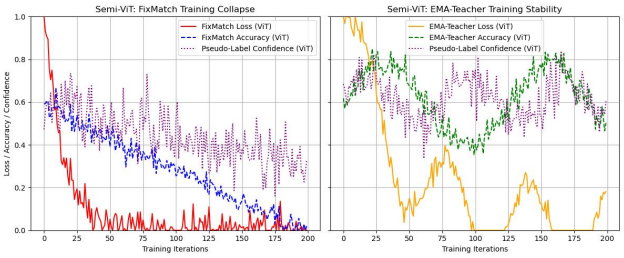


Figure 1. Semi-ViT-Small trained with only 1% of the labeled data of ImageNet-1K. Left: Training fails when FixMatch is used. The loss decreases but test accuracy remains near zero. Right: Training is unstable and fails when EMA-Teacher is used: The loss and accuracy oscillate heavily.

Based on the previous discussions, this paper integrates ViM into an SSL framework for classification and proposes a ViM-based SSL model, hereinafter called Semi-ViM. Semi-ViM incorporates two key innovations aimed at improving model stability during training and performance: LyapEMA and SSMixup. LyapEMA is a novel parameter update mechanism that leverages the Lyapunov theory to dynamically adjust parameter updates, ensuring a more stable training process. LyapEMA addresses the training failures encountered when using only a small amount of labeled

*Corresponding author.

data to train Semi-ViT-Small, as shown in Fig 1. LyapEMA also helps prevent training collapse, commonly encountered when using the exponential moving average (EMA) and FixMatch parameter update mechanisms [4]. SSMixup, on the other hand, is a novel mixup strategy that operates at the hidden state space of SSMs and goes beyond the traditional input-level mixup strategy. Specifically, SSMixup incorporates mixed hidden states as residual units in the training process, improving the model's capability to generalize under sparse label conditions. Our main contributions can be summarized as follows:

- We propose Semi-ViM, an SSL framework that outperforms other SSL methods based on CNNs and Transformers. The number of parameters of Semi-ViM is only around one-fourth of that of Semi-ViT.
- We propose LyapEMA, a novel parameter update mechanism that successfully addresses the training instability issues commonly found in Semi-ViT, thus improving model convergence and generalization capabilities.
- We propose SSMixup, a hidden state space-level mixup strategy that effectively enhances the learning capability of bidirectional SSMs.
- Based on evaluations on the the long-tailed ImageNet-LT dataset, we demonstrate Semi-ViM's strong capabilities in mitigating learning biases caused by label imbalance.

2. Related Work

Imbalanced datasets are those in which the different classes are distributed unequally, often resulting in long-tailed distributions [7]. SSL aims to leverage a small amount of labeled data and a large amount of unlabeled data to improve performance, which is useful to deal with imbalanced datasets. SSL has achieved significant advancements in several computer vision tasks [10, 11, 21, 38, 45].

Traditional SSL methods, e.g., entropy minimization and consistency regularization [13, 15], exploit unlabeled data to reduce dependence on manual annotation. For example, consistency regularization leverages unlabeled data under the assumption that a model's predictions remain consistent under different input perturbations. Mainstream methods include Π-Model [22], Mean Teacher [40], Virtual Adversarial Training (VAT) [30], and FixMatch [38], which is an important benchmark in SSL [38]. Pseudo-Labeling [23] is a common strategy used in SSL. It consists of using the model's predictions on unlabeled data as *pseudo-labels* for training. Noisy Student, FixMatch [38], and FlexMatch [45] adopt this strategy and use confidence thresholds to filter out pseudo-labels.

Several SOTA SSL methods are based on FixMatch or any of its variants, e.g., FlexMatch and CoMatch [24], which integrate contrastive learning or self-supervised methods to enhance performance. However, these methods typically assume that the class distribution is balanced, which is rarely the case in real-world scenarios [12]. When dealing with imbalanced datasets [8], these SOTA SSL methods face two primary challenges. First, pseudo-label biases, i.e., models tend to predict more frequently the majority classes when generating pseudo-labels, exacerbating the imbalance issue and limiting learning from the minority classes [19]. Second, low-quality pseudo-labels, i.e., models tend to assign low confidence scores to those pseudo-labels generated for the minority class samples, thus reducing their impact on training and potentially leading to training collapse [4]. For example, FixMatch relies on a fixed confidence threshold for selecting pseudo-labels, but in an imbalanced dataset setting, pseudo-labels of the minority classes often fail to reach this threshold, leading to insufficient learning from such classes [4, 43].

3. Preliminaries

State Space Models and ViM: SSMs have a long-standing history in control systems and signal processing, offering a structured approach to sequential data modeling [47]. Deep learning adopts SSMs to more effectively handle long-range dependencies compared to traditional recurrence-based models. Unlike Transformers, which rely on global self-attention, SSMs leverage structured state transitions for sequential data propagation, enabling subquadratic computational complexity and linear memory scaling [47]. In computer vision, SSMs have been integrated into deep architectures to enhance representation learning by modeling both spatial and temporal dependencies [27, 47]. The SSM is formulated as a discrete-time state-space system [47]:

$$h_t = Ah_{t-1} + Bx_t, \quad y_t = Ch_t, \tag{1}$$

where $A \in \mathbb{R}^{N\times N}$ is the state transition matrix; $B \in \mathbb{R}^{N\times 1}$ and $C \in \mathbb{R}^{1\times N}$ are input and output projection matrices, respectively; x_t represents the input sequence at time step t; and y_t is the correspnding output representation. The parameters of an SSM are usually optimized in a data-dependent manner to ensure computational efficiency and parallelization.

The bidirectional extension of SSMs further improves feature learning by capturing both forward and backward information, making it a competitive alternative to self-attention mechanisms used by Transformers. [47]. The bidirectional SSM operates in the forward (^{f}) and backward (^{b}) direction:

$$h_t^f = Ah_{t-1}^f + Bx_t, \quad y_t^f = Ch_t^f; \tag{2}$$

$$h_t^b = Ah_{t+1}^b + Bx_t, \quad y_t^b = Ch_t^b. \tag{3}$$

The final output at time step t is obtained by merging the forward and backward outputs through a gating mechanism:

$$y_t = \text{SiLU}(z) \odot y_t^f + \text{SiLU}(z) \odot y_t^b, \tag{4}$$

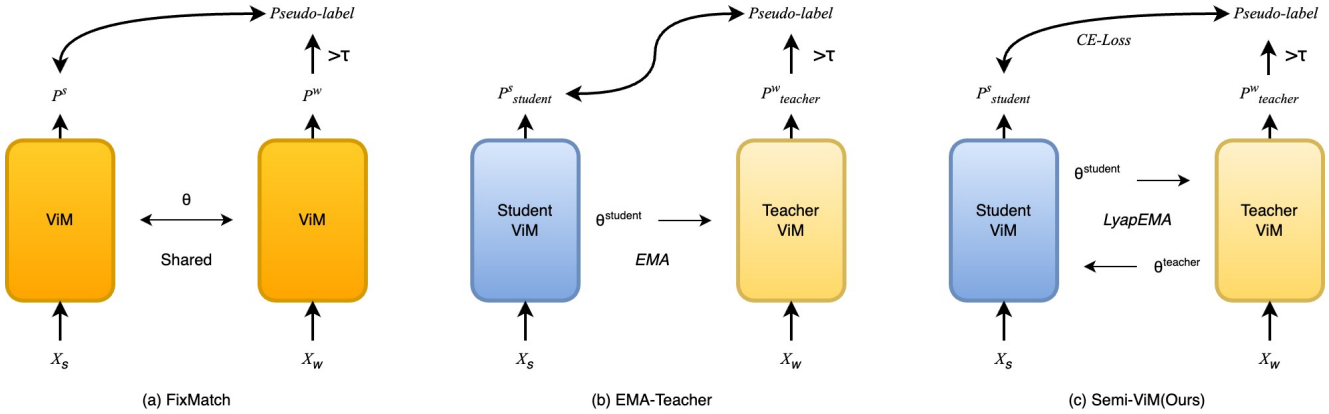


Figure 2. Differences between (a) FixMatch, (b) EMA-Teacher and (c) Semi-ViM (ours). In the figure, X_s and X_w are, respectively, strongly and weakly augmented inputs; p^s and $p^s_{student}$ are the predictions of the student model on X_s; p^w and $p^w_{teacher}$ are the predictions of the teacher model on X_w; τ is the threshold used to select pseudo-labels; and $\theta, \theta^{student}, \theta^{teacher}$ are, respectively, the learnable parameters for the entire model, the student model, and the teacher model.

where SiLU$(\cdot)$ is the Sigmoid-weighted Linear Unit activation function, and z is a gating vector that adaptively balances bidirectional information.

ViM adopts a bidirectional SSM for visual representation learning and integrates positional embeddings for spatial awareness. In ViM, the input image is first partitioned into patches and projected onto a high-dimensional feature space before being processed. With a linear computational complexity, ViM provides data-dependent global visual modeling, significantly improving efficiency in high-resolution tasks compared to Transformer-based architectures [26, 47].

Mixup strategies: These strategies interpolate input samples and labels, and hence can be used as data augmentation techniques to improve generalization capabilities [2]. In SSL, they can enhance pseudo-labeling and provide consistency regularization [34]. Recent variants extend mixup strategies to the feature space and pseudo-label frameworks, e.g., the probabilistic pseudo mixup strategy refines pseudo-label quality by weighting interpolated samples based on their confidence scores, which are derived from the model's softmax outputs or uncertainty estimates. These variants can improve SSL robustness, particularly for ViTs, which benefit from stronger regularization [4].

Weak and strong data augmentation: Weak data augmentation involves simple transformation, e.g., random cropping, flipping, or slight color jitter, which introduce small variations to the original data to help a model learn from basic invariances. Strong data augmentation, on the other hand, applies more aggressive transformations, e.g., large crops, strong color distortions, or more significant perturbations, which force a model to learn more robust representations by simulating more challenging scenarios [31].

4. Proposed Semi-ViM framework

Semi-ViM adopts a teacher-student framework and uses a labeled dataset, $\mathcal{X}_l = \{(x_i^l, y_i^l)\}_{i=1}^{N_l}$, and an unlabeled dataset $\mathcal{X}_u = \{x_i^u\}_{i=1}^{N_u}$ for training. Specifically, the teacher model, $f_{\text{teacher}}(\theta^{\text{teacher}})$, generates pseudo-labels, $\{\hat{y}_i\}$, for unlabeled samples, $\{x_i^u\}$, and outputs confidence scores, $\{o_i\}$, for the pseudo-labels. The student model, $f_{\text{student}}(\theta^{\text{student}})$, is then trained on samples mixed according to their confidence scores. The difference between the training paradigm of Semi-ViM and those of FixMatch and EMA-Teacher is shown in Fig 2.

In this section, we first introduce LyapEMA, which is inspired by Lyapunov stability theory [36] and stabilizes parameter updates, thus preventing model collapse during the training of Semi-ViM. Next, we introduce SSMixup, a new mixup strategy designed for SSM-based models. Unlike traditional mixup strategies that operate solely on the input level, SSMixup mixes the hidden states of bidirectional SSMs. Finally, we detail the training and loss function of Semi-ViM.

4.1. LyapEMA

LyapEMA is a novel optimization framework that integrates Lyapunov stability theory with EMA. LyapEMA is designed for teacher-student frameworks under an SSL setting. It has two main objectives. The first is to ensure stable parameter updates, because training SSL frameworks, like Semi-ViT, with a limited amount of labeled data can lead to training collapse. The second objective is reducing the learning bias associated with pseudo-labels while preserving the advantages of EMA in refining the teacher model.

Let θ^{student} and θ^{teacher} denote the student and teacher model parameters, respectively. EMA updates the teacher

model using gradient descent as follows:

$$\theta_t^{\text{teacher}} = \alpha_t \theta_{t-1}^{\text{teacher}} + (1-\alpha_t)\theta_t^{\text{student}}, \quad (5)$$

where $\alpha \in (0,1)$ is the EMA decay factor. However, EMA cannot guarantee stability, potentially leading to divergence due to accumulated bias associated with learning from pseudo-labels. To address this issue, LyapEMA introduces Lyapunov stability constraints to guide the updates. Let us define a Lyapunov candidate function, $V(\theta^{\text{student}}, \theta^{\text{teacher}})$, to measure the discrepancy between the student and teacher models:

$$V(\theta^{\text{student}}, \theta^{\text{teacher}}) = \frac{1}{2}\|\theta^{\text{student}} - \theta^{\text{teacher}}\|_2^2. \quad (6)$$

Note that the function in Eq. 6 acts as a stability indicator, ensuring that the student model does not drift too far from the teacher model. The Lyapunov stability criterion requires that the derivative of $V(\cdot,\cdot)$ w.r.t. time, i.e., $\dot{V}$, satisfies:

$$\dot{V} = \frac{dV}{dt} = (\theta^{\text{student}} - \theta^{\text{teacher}})^{\text{T}}(\dot{\theta}^{\text{student}} - \dot{\theta}^{\text{teacher}}) < 0, \quad (7)$$

where $\dot{\theta}^{\text{student}}$ and $\dot{\theta}^{\text{teacher}}$ also derivatives w.r.t. time. The criterion in Eq. 7 guarantees that the discrepancy between the student and teacher models decreases over time, ensuring convergence.

To enforce stability while leveraging EMA for smooth teacher model updates, LyapEMA updates the student model parameters by using aLyapunov regularization term:

$$\theta_t^{\text{student}} = \theta_{t-1}^{\text{student}} - \eta\nabla_\theta \mathcal{L}(\theta^{\text{student}}) + \lambda(\theta_t^{\text{teacher}} - \theta_t^{\text{student}}), \quad (8)$$

where $\mathcal{L}(\theta^{\text{student}})$ is a semi-supervised loss function, η is the learning rate, and λ is a stability coefficient that ensures the student model remains close to the teacher model.

LyapEMA updates the teacher model parameters by using EMA as denoted by Eq. 5. However, the EMA decay factor α_t is dynamically adjusted based on the Lyapunov function:

$$\alpha_t = \text{sigmoid}(\gamma\|\theta_t^{\text{student}} - \theta_{t-1}^{\text{teacher}}\|_2), \quad (9)$$

where γ is a sensitivity hyperparameter controlling how α_t adapts based on the student-teacher discrepancy, and the sigmoid function ensures α_t remains within $(0,1)$, preventing instability in the updates.

LyapEMA ensures that updates lead to stable training by using the *Lyapunov decrease condition*:

$$V_{t+1} - V_t = \frac{1}{2}\|\theta_{t+1}^{\text{student}} - \theta_{t+1}^{\text{teacher}}\|_2^2 - \frac{1}{2}\|\theta_t^{\text{student}} - \theta_t^{\text{teacher}}\|_2^2. \quad (10)$$

Algorithm 1 LyapEMA

Input: Initial parameters $\theta^{\text{student}}, \theta^{\text{teacher}}$, learning rate η, stability coefficient λ, sensitivity γ
Output: Stabilized parameters $\theta^{\text{student}}, \theta^{\text{teacher}}$

while training not converged **do**
  Compute loss $\mathcal{L}(\theta^{\text{student}})$
  Update student model: $\theta_t^{\text{student}} = \theta_{t-1}^{\text{student}} - \eta\nabla_\theta\mathcal{L}(\theta^{\text{student}}) + \lambda(\theta_t^{\text{teacher}} - \theta_t^{\text{student}})$
  Compute adaptive EMA decay in LyapEMA: $\alpha_t = \text{sigmoid}(\gamma\|\theta_t^{\text{student}} - \theta_{t-1}^{\text{teacher}}\|_2)$
  Update teacher model: $\theta_t^{\text{teacher}} = \alpha_t\theta_{t-1}^{\text{teacher}} + (1-\alpha_t)\theta_t^{\text{student}}$
  Ensure stability: Verify $V(\theta)$ decreases
  if $V_t \geq V_{t-1}$ **then**
    Reduce learning rate: $\eta \leftarrow \eta/2$
    Increase stability coefficient: $\lambda \leftarrow \lambda + \Delta\lambda$
    Recompute student and teacher updates
  end if
end while
return $\theta^{\text{teacher}}, \theta^{\text{student}}$

By using the LyapEMA update rules in Eq. 10 and expanding, we obtain:

$$V_{t+1} - V_t \approx -\eta\|\nabla_\theta\mathcal{L}(\theta^{\text{student}})\|_2^2 - \lambda\|\theta_t^{\text{student}} - \theta_t^{\text{teacher}}\|_2^2. \quad (11)$$

Since both terms in Eq. 11 are negative, V decreases monotonically, ensuring stability. The complete LyapEMA framework is summarized in Algorithm 1. In Section 5.1, we conduct experiments not only on Semi-ViM but also on other SSL methods to validate LyapEMA's effectiveness.

4.2. SSMixup

SSMixup introduces mixup strategies in the input and the hidden state space of a bidirectional SSM to effectively leverage unlabeled data and enhance generalization in SSL. As mentioned before, Semi-ViM adopts a teacher-student framework where the student model, $f_{\text{student}}(\theta^{\text{student}})$, is trained on the mixed samples.

In SSMixup, the pseudo-label confidence scores, o_i and o_j, of two unlabeled samples, x_i^u and x_j^u, respectively, are used first to compute a mixup coefficient:

$$\psi = \frac{o_i}{o_i + o_j}. \quad (12)$$

The mixup coefficient ψ is used to form the mixed input and labels:

$$x_{\text{mix}} = \psi\, x_i^u + (1-\psi)\, x_j^u, \quad y_{\text{mix}} = \psi\, \hat{y}_i + (1-\psi)\, \hat{y}_j. \quad (13)$$

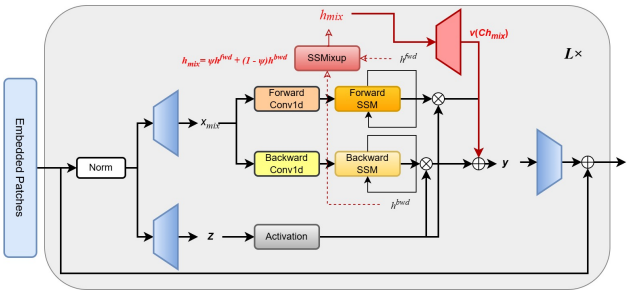


Figure 3. The SSMixup strategy. The red part applies mixup at the hidden state level. L denotes to the number of SSMixup encoder layers in Semi-ViM. It is a hyperparameter typically set to 24.

The mixed input x_{mix} is then fed into the forward and backward components of the bidirectional SSM, yielding hidden states:

$$h^{\text{fwd}} = \text{SSM}_{\text{fwd}}(x_{\text{mix}}), \quad h^{\text{bwd}} = \text{SSM}_{\text{bwd}}(x_{\text{mix}}). \tag{14}$$

SSMixup mixes the hidden states by using the mixup coefficient, ψ:

$$h_{\text{mix}} = \psi\, h^{\text{fwd}} + (1-\psi)\, h^{\text{bwd}}, \tag{15}$$

where h_{mix} is computed as a residual unit and C is defined in Eq. 1. The forward and backward outputs of the bidirectional SSM are merged to compute the final output (see Eq. 4):

$$y_t = \text{SiLU}(z) \odot y_t^f + \text{SiLU}(z) \odot y_t^b + \nu(Ch_{\text{mix}}). \tag{16}$$

where ν is the control coefficient. The computed y_t is used in a cross-entropy loss with y_{mix} to update θ^{student}. Parameters θ^{teacher} are smoothly updated via LyapEMA, maintaining stable pseudo-label generation.

Figure 3 illustrates the functionality of SSMixup. By introducing mixup strategies at both the input (see Eq. 13) and hidden state levels (see Eq. 15), combined with confidence-driven weighting, SSMixup effectively exploits unlabeled data and significantly improves generalization capabilities in SSL settings.

4.3. Training Semi-ViM

Training Semi-ViM involves three main steps. The first step involves self-supervised pre-training based on Autoregressive Mamba (ARM) [33], where labeled data is treated as unlabeled data and used as part of the training samples. The second step involves supervised fine-tuning using only labeled data. The third step involves SSL. These three steps are detailed next.

Self-supervised pre-training based on ARM: To improve feature extraction, we propose to leverage ARM for self-supervised pre-training of Semi-ViM. ARM is better suited for ViM pre-training than Masked Autoencoders (MAEs) and DINO-V2 (contrastive learning) [32] due to its alignment with ViM's sequential state-space modeling [33]. Unlike MAEs, which rely on masked image reconstruction and bidirectional Transformers, Semi-ViM's bidirectional state-space modeling allows to effectively leverage ARM's autoregressive token prediction, making ARM a more natural fit [17]. Compared to DINO-V2, which depends on global feature alignment and large-batch contrastive learning, ARM directly optimizes sequential token dependencies, avoiding instability from batch size sensitivity and augmentation complexity [9]. By leveraging ViM's structured recurrence, ARM enables faster training, better scalability, and improved efficiency over both methods. These statements are validated by our experiments in Section 5.

Fully-supervised fine-tuning: After completing ARM-based pre-training, Semi-ViM undergoes fine-tuning using only labeled data. This process aims to leverage the representations learned during pre-training to adapt the model to specific downstream tasks. During fine-tuning, most pre-trained parameters remain unchanged, with only certain layers or global parameters being updated to a limited extent in order to improve performance in SSL. Fine-tuning specifically targets layers responsible for learning bidirectional dependencies, i.e., the forward and backward SSMs are updated to better capture dependencies specific to a classification task. The class token used in image classification tasks is also fine-tuned during this phase.

SSL: In the third training step, we use LyapEMA, in conjunction with SSMixup to fully exploit unlabeled data while ensuring training stability and efficiency. This last training step consists of five key components: weak and strong data augmentation, pseudo-label generation, LyapEMA, SSMixup, and minimization of the loss function.

For each unlabeled sample x^u, we generate a weakly augmented sample x_w^u and a strongly augmented sample x_s^u. The weakly augmented sample x_w^u is processed by the teacher model f_{teacher} to produce a set of predicted probabilities, one for each class:

$$p_{\text{teacher}_{x_w^u}} = f_{\text{teacher}}(x_w^u). \tag{17}$$

The pseudo-label for x_w^u is then determined as the class associated with the highest probability in set $p_{\text{teacher}_{x_w^u}}$:

$$\hat{y}^u = \arg\max(p_{\text{teacher}_{x_w^u}}). \tag{18}$$

Note that only unlabaled samples with a class probability exceeding the threshold τ are retained for training; i.e., only those with high-score pseudo-labels are retained. SSMixup is then applied to these unlabeled samples with high-score pseudo labels in order to be used to train the student model.

Variant name	Backbone	Param.	Input size
Semi-ViM-Tiny	ViM-Tiny [47]	7M	224^2
Semi-ViM-Small	ViM-Small [47]	26M	224^2
Semi-ViM-Base	ViM-Base [47]	98M	224^2

Table 1. The different variants of Semi-ViM.

The overall loss function consists of the supervised loss L_{labeled} and the unsupervised loss $L_{\text{unlabeled}}$:

$$L = L_{\text{labeled}} + \mu L_{\text{unlabeled}}. \tag{19}$$

The supervised loss, L_{labeled}, is based on the standard cross-entropy (CE) loss using labeled samples:

$$L_{\text{labeled}} = \frac{1}{N_l} \sum_{i=1}^{N_l} \text{CE}(f_{\text{student}}(x_i^l), y_i^l). \tag{20}$$

The unsupervised loss $L_{\text{unlabeled}}$ uses strongly augmented samples, x_s^u as follows:

$$L_{\text{unlabeled}} = \frac{1}{N_u} \sum_{i=1}^{N_u} \mathbb{1}(o_i > \tau) \text{CE}(f_{\text{student}}(x_{s,i}^u), \hat{y}_i^u), \tag{21}$$

where the indicator function $\mathbb{1}(o > \tau)$ selects only those unlabeled samples with high-score pseudo-labels, where o denotes the probability of the class assocated with the pseduo-label. Finally, the teacher model is updated using LyapEMA to stabilize parameter evolution and improve the quality of pseudo-labels.

5. Experiments

Table 1 lists the different variants of Semi-ViM we use in our experiments: Tiny, Small, and Base. These variants are trained on ImageNet-1K using 8 NVIDIA A100 (80GB) GPUs with DistributedDataParallel (DDP) PyTorch. We use AdamW as the optimizer with a weight decay of 0.05, and a learning rate that follows a cosine annealing schedule starting from 5×10^{-4} with 10 epochs of warm-up. We use 300 epochs for training with a global batch size of 1024 (128 per GPU) and automatic mixed precision (AMP) enabled. Strong data augmentation includes AutoAugment, Mixup ($\alpha = 0.2$), CutMix ($\alpha = 1.0$), and label smoothing ($\varepsilon = 0.1$). The total training time is approximately 70 hours.

We conduct classification experiments on ImageNet-1K [9] and ImageNet-LT [28]. ImageNet-1K contains 1.28M training and 50K validation images across 1,000 classes. ImageNet-LT has a long-tailed class distribution. We use a confidence threshold of $\tau = 0.7$ for pseudo-label selection, with weak and strong data augmentations applied to unlabeled data. Inference is performed using a 224×224 center crop on the images.

5.1. Results and analysis

Semi-ViM demonstrates superior performance across multiple experimental settings as tabulated in Table 2. On ImageNet-1K, Semi-ViM-Base achieves 85.40% accuracy with only 10% of the labeled data, outperforming Semi-ViT-Huge. Compared to SemiFormer, Semi-ViM consistently excels, confirming the suitability of ViM for SSL.

For the more challenging ImageNet-LT dataset, Semi-ViM-Base achieves 77.40% accuracy with only 10% of the labeled data, significantly surpassing Semi-ViT-Huge and SemiFormer. This demonstrates Semi-ViM's adaptability to long-tailed data distributions and its capability to work with rare category representations. With only 1% of the labeled data, Semi-ViM-Base achieves 66.30% accuracy, surpassing Semi-ViT-Huge and demonstrating its strength in SSL.

In terms of the Top-5 accuracy on ImageNet-LT with only 1% of the labeled data (last column of Table 2), Semi-ViM-Base outperforms Semi-ViT-Huge by 6.17% and CNN-based methods, like FixMatch and PAWS, by over 15%, highlighting its superior feature representation, class diversity learning, and robustness against data imbalance.

Table 3 tabulates the Top-1 Average Precision (AP) on ImageNet-LT of several fully-supervised models trained exclusively with labeled data. Semi-ViM-Base with just 10% of the labeled data achieves competitive accuracy. This suggests that our framework efficiently extracts meaningful representations even with significantly fewer labeled samples, making it highly suitable for real-world scenarios where labeled data may be scarce.

Figure 4 graphically summarizes the performance of Semi-ViM and that of other SOTA SSL models and fully-supervised models.

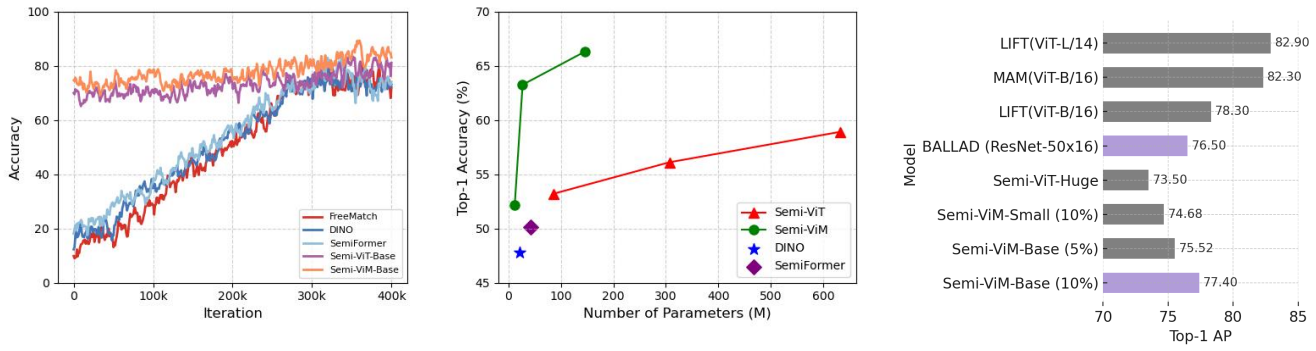
5.2. Ablation study

Several pre-training strategies, parameter updating methods, and mixup strategies are compared in Table 4, using Semi-ViM-Base. ARM pre-training consistently achieves the best performance, demonstrating its effectiveness in leveraging sequential dependencies inherent to the ViM. ARM reaches 81.90% (1% of the labeled data) and 85.40% (10% of the labeled data) when used in conjunction with LyapEMA+SSMixup, significantly outperforming MAEs and DINO-V2 across all settings. This confirms that ARM, designed specifically for sequential state-space models, is a more appropriate fit for ViM than contrastive learning (DINO-V2) or MAEs.

In terms of parameter updating methods, LyapEMA outperforms EMA in all configurations, particularly when used in conjunction with ARM. Unlike EMA, LyapEMA stabilizes updates by leveraging Lyapunov theory, preventing pseudo-label collapse and ensuring more reliable parameter evolution. This is particularly beneficial in low-labeled data scenarios, where training instability is a key challenge.

CNN-based method	ImageNet-1K					ImageNet-LT				
	Backbone	Param.	1%	5%	10%	Para.m	1%	5%	10%	Top5(1%)
FixMatch [38]	ResNet-50	25M	-	67.83	71.74	25M	-	52.32	57.68	-
PAWS [1]	ResNet-50	27M	66.54	69.21	76.55	27M	46.30	56.13	61.81	55.19
FreeMatch [42]	ResNet-50	26M	55.42	70.11	74.50	26M	-	55.54	60.94	-
SoftMatch [5]	ResNet-50	28M	-	70.40	76.20	28M	-	57.41	61.02	-
SimCLRv2(self-distilled) [6]	ResNet-152	28M	76.65	78.23	80.90	28M	48.55	59.30	62.10	67.28
SimCLRv2(distilled) [6]	ResNet-50	28M	75.90	77.40	80.20	28M	47.70	57.23	60.81	68.50
Co-training method	ImageNet-1K					ImageNet-LT				
	Backbone	Param.	1%	5%	10%	Para.m	1%	5%	10%	Top5(1%)
Co-Training [35]	CT(MLP)	56M	80.03	81.65	81.70	56M	58.62	64.79	71.20	78.30
MCT [35]	MCT (MLP)	176M	80.75	82.08	85.20	176M	63.18	72.60	75.30	80.11
Transformer-based method	ImageNet-1K					ImageNet-LT				
	Backbone	Param.	1%	5%	10%	Param.	1%	5%	10%	Top-5 (1%)
DINO [14]	ViT-Small	22M	65.21	71.45	72.53	22M	47.80	55.30	59.90	64.25
SemiFormer [43]	ViT-S+Conv	42M	-	72.80	73.28	42M	50.10	58.00	63.50	67.87
DPT [44]	ViT-Huge	604M	80.20	82.50	84.06	604M	55.91	64.76	68.15	75.40
Semi-ViT-Base [4]	ViT-Base	86M	71.30	77.52	79.90	86M	53.20	62.90	68.40	71.50
Semi-ViT-Large [4]	ViT-Large	307M	76.81	80.50	82.26	307M	56.80	66.10	71.20	74.70
Semi-ViT-Huge [4]	ViT-Huge	632M	80.47	83.20	84.27	632M	58.90	68.40	73.50	76.90
REACT [25]	ViT-Huge	651M	79.13	83.23	84.74	632M	59.36	68.71	73.08	77.45
Semi-ViM-Tiny	ViM-Tiny	12M	76.10	78.30	81.00	12M	52.13	62.90	68.30	71.11
Semi-ViM-Small	ViM-Small	27M	80.30	81.40	83.10	27M	63.24	72.15	74.68	80.02
Semi-ViM-Base	**ViM-Base**	**146 M**	**81.90**	**83.25**	**85.40**	**146M**	**66.30**	**75.52**	**77.40**	**83.07**

Table 2. Accuracy (%) of the different variants of Semi-ViM (in pink) and other SOTA SSL models over 5 folds when 1%, 5% and 10% of the labeled data is used for training. The best performance of other SOTA models is highlighted in peach.



(a) (b) (c)

Figure 4. (a) Training accuracy (%) on ImageNet-1K of Semi-ViM-Base and other SOTA SSL methods using only 10% of the labeled data. (b) Top-1 accuracy (%) on ImageNet-LT (1% of the labeled data) of Semi-ViM and other SOTA SSL algorithms across different model scales. (c) Top-1 Average Precision (%) on ImageNet-LT of Semi-ViM-Base and several fully supervised models.

Regarding mixup strategies, SSMixup provides the largest performance boost, outperforming ProbMixup consistently across all pre-training methods. With ARM, LyapEAM+SSMixup boosts accuracy by 2-3% compared to ProbMixup. This confirms that SSMixup is highly effective for SSL, mitigating pseudo-label noise and improving generalization in low-data settings. The combination of ARM pre-training, and LyapEMA+SSMixup achieves the highest accuracy, underscoring the importance of pre-training alignment, stable parameters updates, and structured aug-

Model	Labeled data	Top-1 AP
LIFT(ViT-L/14) [37]	100%	82.90
MAM(ViT-B/16) [18]	100%	82.30
LIFT(ViT-B/16) [37]	100%	78.30
BALLAD (ResNet-50) [29]	100%	76.50
Semi-ViT-Huge [4]	10%	73.50
Semi-ViM-Small	**10%**	**74.68**
Semi-ViM-Base	**5%**	**75.52**
Semi-ViM-Base	**10%**	**77.40**

Table 3. Top-1 AP (%) of fully-supervised models and the variants of Semi-ViM on ImageNet-LT. Semi-ViM-Base (in purple), using only 10% of the labeled data, outperforms BALLAD (ResNet-50×16).

Pretrained	Method	1%	10%
None	finetune	-	-
	EMA	42.11	47.41
	LyapEMA	44.73	53.26
	+ProbMixup	44.78	49.57
	+SSMixup	**45.29**	**54.73**
MAE	finetune	63.05	73.14
	EMA	71.30	74.20
	LyapEMA	65.30	75.65
	+ProbMixup	**66.09**	75.87
	+SSMixup	66.02	**76.15**
DINO-V2	finetune	62.02	72.63
	EMA	63.45	73.57
	LyapEMA	64.84	75.21
	+ProbMixup	65.53	74.04
	+SSMixup	**65.32**	**76.07**
ARM	finetune	66.53	74.21
	EMA	72.53	79.90
	LyapEMA	76.02	78.03
	+ProbMixup	79.83	80.52
	+SSMixup	**81.90**	**85.40**

Table 4. Top-1 accuracy (%) on ImageNet-1K of Semi-ViM-Base under different pre-training strategies and parameter updating methods using 1% and 10% of the labeled data for training. "+" means the mixup strategy used in conjunction with LyapEMA. Best results are highlighted in **bold** font.

mentation in semi-supervised models based on ViM.

Table 5 compares different parameter updating methodos, demonstrating the superior stability and generalization capability of LyapEMA in SSL. The EMA-Teacher suffers from training instability, particularly in Semi-ViT-Small and Semi-ViM-Small with only 1% of the labeled data, where training fails to converge. In contrast, LyapEMA effectively prevents collapse, achieving 61.30% and 80.30% accuracy, respectively, showing its robustness in low-labeled data scenarios. By dynamically adapting the EMA decay factor based on the student-teacher discrepancy, LyapEMA mitigates pseudo-label bias and enhances convergence. Additional ablation studies and the model's performance under our designed *extreme imbalance scenarios* are provided in the supplementary material.

Model	Method	1%	10%
FreeMatch [42]	FixMatch [38]	–	–
	EMA-Teacher [3]	55.42	74.50
	LyapEMA	56.70	75.07
Semi-ViT-S [4]	FixMatch [38]	–	×
	EMA-Teacher [3]	×	75.80
	LyapEMA	61.30	76.60
Semi-ViT-B [4]	FixMatch [38]	–	×
	EMA-Teacher [3]	66.41	78.15
	LyapEMA	68.84	80.43
Semi-ViM-S	FixMatch [38]	×	×
	EMA-Teacher [3]	×	74.01
	LyapEMA	80.30	85.40
Semi-ViM-B	FixMatch [38]	×	71.28
	EMA-Teacher [3]	70.53	76.11
	LyapEMA	81.90	85.40

Table 5. Top-1 accuracy (%) on ImageNet-1K of FixMatch, EMA-Teacher and LyapEMA with different models using 1% and 10% of the labeled data for training. "×" means the training failed (accuracy near 0). "–" means no result is available. Results in light blue indicate risk of training instability and collapse.

6. Conclusion

In this paper, we introduced Semi-ViM, a novel semi-supervised learning framework that leverages bidirectional SSMs to mitigate learning bias caused by label imbalance in classification tasks. By integrating LyapEMA, a novel stability-aware parameter update mechanism, and SS-Mixup, a hidden-state-level mixing strategy, Semi-ViM enhances both training stability and feature learning. Our extensive experiments on ImageNet-1K and ImageNet-LT demonstrate that Semi-ViM significantly outperforms CNN-based and Transformer-based SSL models, achieving superior accuracy with fewer labeled samples. Moreover, Semi-ViM requires far fewer parameters than ViT-based SSL models. These findings highlight the potential of state space models as a competitive alternative to Transformers in semi-supervised learning, particularly in addressing real-world imbalanced datasets.

References

[1] Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Armand Joulin, Nicolas Ballas, and Michael Rabbat. Semi-supervised learning of visual features by non-parametrically predicting view assignments with support samples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8443–8452, 2021. 7

[2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32, 2019. 3

[3] Zhaowei Cai, Avinash Ravichandran, Subhransu Maji, Charless Fowlkes, Zhuowen Tu, and Stefano Soatto. Exponential moving average normalization for self-supervised and semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 194–203, 2021. 8

[4] Zhaowei Cai, Avinash Ravichandran, Paolo Favaro, Manchen Wang, Davide Modolo, Rahul Bhotika, Zhuowen Tu, and Stefano Soatto. Semi-supervised vision transformers at scale. *Advances in Neural Information Processing Systems*, 35:25697–25710, 2022. 1, 2, 3, 7, 8

[5] Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xing Xie, Bhiksha Raj, and Marios Savvides. Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. *arXiv preprint arXiv:2301.10921*, 2023. 7

[6] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020. 7

[7] Nieves Crasto. Class imbalance in object detection: An experimental diagnosis and study of mitigation strategies. *arXiv preprint arXiv:2403.07113*, 2024. 2

[8] Charika De Alvis and Suranga Seneviratne. A survey of deep long-tail classification advancements. *arXiv preprint arXiv:2404.15593*, 2024. 2

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5, 6

[10] Yue Duan, Zhen Zhao, Lei Qi, Luping Zhou, Lei Wang, and Yinghuan Shi. Towards semi-supervised learning with non-random missing labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16121–16131, 2023. 2

[11] Yue Duan, Zhen Zhao, Lei Qi, Luping Zhou, Lei Wang, and Yinghuan Shi. Roll with the punches: Expansion and shrinkage of soft label selection for semi-supervised fine-grained learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11829–11837, 2024. 2

[12] Marawan Elbatel, Hualiang Wang, Jixiang Chen, Hao Wang, and Xiaomeng Li. Learning unlabeled clients divergence via anchor model aggregation for federated semi-supervised learning. *arXiv e-prints*, pages arXiv–2407, 2024. 2

[13] Yue Fan, Anna Kukleva, Dengxin Dai, and Bernt Schiele. Revisiting consistency regularization for semi-supervised learning. *International Journal of Computer Vision*, 131(3): 626–643, 2023. 2

[14] Enrico Fini, Pietro Astolfi, Karteek Alahari, Xavier Alameda-Pineda, Julien Mairal, Moin Nabi, and Elisa Ricci. Semi-supervised learning made simple with self-supervised clustering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3187–3197, 2023. 7

[15] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17, 2004. 2

[16] Lan-Zhe Guo, Lin-Han Jia, Jie-Jing Shao, and Yu-Feng Li. Robust semi-supervised learning in open environments. *Frontiers of Computer Science*, 19(8):198345, 2025. 1

[17] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 5

[18] Ahmet Iscen, Alireza Fathi, and Cordelia Schmid. Improving image recognition by retrieving from web-scale image-text data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19295–19304, 2023. 8

[19] Zhen Jiang, Lingyun Zhao, Yu Lu, Yongzhao Zhan, and Qirong Mao. A semi-supervised resampling method for class-imbalanced learning. *Expert Systems with Applications*, 221:119733, 2023. 2

[20] Suraj Kothawade, Pavan Kumar Reddy, Ganesh Ramakrishnan, and Rishabh Iyer. Basil: Balanced active semi-supervised learning for class imbalanced datasets. *arXiv preprint arXiv:2203.05651*, 2022. 1

[21] Alex Kurakin, Colin Raffel, David Berthelot, Ekin Dogus Cubuk, Han Zhang, Kihyuk Sohn, and Nicholas Carlini. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. 2020. 2

[22] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016. 2

[23] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, page 896. Atlanta, 2013. 2

[24] Junnan Li, Caiming Xiong, and Steven CH Hoi. Comatch: Semi-supervised learning with contrastive graph regularization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9475–9484, 2021. 2

[25] Haotian Liu, Kilho Son, Jianwei Yang, Ce Liu, Jianfeng Gao, Yong Jae Lee, and Chunyuan Li. Learning customized visual models with retrieval-augmented knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15148–15158, 2023. 7

[26] Xiao Liu, Chenxu Zhang, and Lei Zhang. Vision mamba: A comprehensive survey and taxonomy. *arXiv preprint arXiv:2405.04404*, 2024. 3

[27] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan

Liu. Vmamba: Visual state space model. *Advances in neural information processing systems*, 37:103031–103063, 2025. 2

[28] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2537–2546, 2019. 6

[29] Teli Ma, Shijie Geng, Mengmeng Wang, Jing Shao, Jiasen Lu, Hongsheng Li, Peng Gao, and Yu Qiao. A simple long-tailed recognition baseline via vision-language model. *arXiv preprint arXiv:2111.14745*, 2021. 8

[30] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018. 2

[31] Khanh-Binh Nguyen. Sequencematch: revisiting the design of weak-strong augmentations for semi-supervised learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 96–106, 2024. 3

[32] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 5

[33] Sucheng Ren, Xianhang Li, Haoqin Tu, Feng Wang, Fangxun Shu, Lei Zhang, Jieru Mei, Linjie Yang, Peng Wang, Heng Wang, et al. Autoregressive pretraining with mamba in vision. *arXiv preprint arXiv:2406.07537*, 2024. 5

[34] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *arXiv preprint arXiv:2101.06329*, 2021. 3

[35] Jay C Rothenberger and Dimitrios I Diochnos. Meta co-training: Two views are better than one. In *2025 Joint Mathematics Meetings (JMM 2025)*. AMS. 7

[36] Shankar Sastry and Shankar Sastry. Lyapunov stability theory. *Nonlinear systems: analysis, stability, and control*, pages 182–234, 1999. 3

[37] Jiang-Xin Shi, Tong Wei, Zhi Zhou, Jie-Jing Shao, Xin-Yan Han, and Yu-Feng Li. Long-tail learning with foundation model: Heavy fine-tuning hurts. In *Forty-first International Conference on Machine Learning*, 2024. 8

[38] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33:596–608, 2020. 2, 7, 8

[39] Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*, 2021. 1

[40] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017. 2

[41] Renzhen Wang, Xixi Jia, Quanziang Wang, Yichen Wu, and Deyu Meng. Imbalanced semi-supervised learning with bias adaptive classifier. *arXiv preprint arXiv:2207.13856*, 2022. 1

[42] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, et al. Freematch: Self-adaptive thresholding for semi-supervised learning. *arXiv preprint arXiv:2205.07246*, 2022. 7, 8

[43] Zejia Weng, Xitong Yang, Ang Li, Zuxuan Wu, and Yu-Gang Jiang. Semi-supervised vision transformers. In *European conference on computer vision*, pages 605–620. Springer, 2022. 1, 2, 7

[44] Zebin You, Yong Zhong, Fan Bao, Jiacheng Sun, Chongxuan Li, and Jun Zhu. Diffusion models and semi-supervised learners benefit mutually with few labels. *Advances in Neural Information Processing Systems*, 36:43479–43495, 2023. 7

[45] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34:18408–18419, 2021. 2

[46] Dingyuan Zhang, Dingkang Liang, Zhikang Zou, Jingyu Li, Xiaoqing Ye, Zhe Liu, Xiao Tan, and Xiang Bai. A simple vision transformer for weakly semi-supervised 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8373–8383, 2023. 1

[47] L Zhu, B Liao, Q Zhang, X Wang, W Liu, and X Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. arxiv 2024. *arXiv preprint arXiv:2401.09417*. 1, 2, 3, 6