

Towards Micro-Action Recognition with Limited Annotations: An Asynchronous Pseudo Labeling and Training Approach

Yan Zhang¹, Lechao Cheng^{1*}, Yaxiong Wang¹, Zhun Zhong^{1,2*}, Meng Wang¹

¹Hefei University of Technology

²University of Nottingham

Abstract

Micro-Action Recognition (MAR) aims to classify subtle human actions in video. However, annotating MAR datasets is particularly challenging due to the subtlety of actions. To this end, we introduce the setting of Semi-Supervised MAR (SSMAR), where only a part of samples are labeled. We first evaluate traditional Semi-Supervised Learning (SSL) methods to SSMAR and find that these methods tend to overfit on inaccurate pseudo-labels, leading to error accumulation and degraded performance. This issue primarily arises from the common practice of directly using the predictions of classifier as pseudo-labels to train the model. To solve this issue, we propose a novel framework, called Asynchronous Pseudo Labeling and Training (APLT), which explicitly separates the pseudo-labeling process from model training. Specifically, we introduce a semi-supervised clustering method during the offline pseudo-labeling phase to generate more accurate pseudo-labels. Moreover, a self-adaptive thresholding strategy is proposed to dynamically filter noisy labels of different classes. We then build a memory-based prototype classifier based on the filtered pseudo-labels, which is fixed and used to guide the subsequent model training phase. By alternating the two pseudo-labeling and model training phases in an asynchronous manner, the model can not only be learned with more accurate pseudo-labels but also avoid the overfitting issue. Experiments on three MAR datasets show that our APLT largely outperforms state-of-the-art SSL methods. For instance, APLT improves accuracy by 14.5% over FixMatch on the MA-12 dataset when using only 50% labeled data. Code is available at <https://github.com/zy-hfut/APLT>.

1 Introduction

Micro-action refers to the fast and tiny movements or body language signals that exhibit by humans during communication [Noroozi *et al.*, 2021], which plays a significant

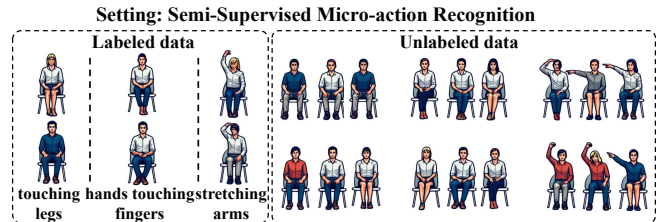


Figure 1: We present a new setting, Semi-Supervised Micro-Action Recognition (SSMAR), which aims to train a model that can recognize subtle, rapid micro-actions in videos by utilizing both labeled and unlabeled data.

role in revealing individuals’ inner states and true intentions. Traditional action recognition [Arnab *et al.*, 2021; Feichtenhofer *et al.*, 2019; Xie *et al.*, 2018; Wang *et al.*, 2024] typically focuses on identifying and classifying overt movements, *e.g.*, running, jumping, etc. In contrast, Micro-Action Recognition (MAR) [Chen *et al.*, 2023b; Liu *et al.*, 2021] targets the detection and analysis of minor movement changes, which are imperceptible to human eyes, making MAR inherently more challenging. Existing MAR methods are mainly developed under a fully-supervised context, which heavily depend on large amounts of high-quality labeled data [Caba Heilbron *et al.*, 2015; Kay *et al.*, 2017; Soomro and others, 2012]. However, acquiring labeled MAR datasets is labor-intensive and costly due to the subtlety of micro-action and privacy concerns. To alleviate the challenges of labeling MAR datasets, we introduce a Semi-Supervised Learning (SSL) setting for MAR, called Semi-Supervised MAR (SSMAR). As shown in Fig. 1, SSMAR aims to harness both labeled and unlabeled data to train a MAR model, enabling performance close to that of a model trained on fully-labeled data.

SSL has been well established in many fields, such as image classification [Sohn *et al.*, 2020; Berthelot *et al.*, 2019], semantic segmentation [Chen *et al.*, 2021; Luo and Yang, 2020], action recognition [Xing *et al.*, 2023; Wu *et al.*, 2023], etc. The popular SSL methods mainly focus on consistent regularization [Rasmus *et al.*, 2015; Ke *et al.*, 2019] and pseudo-labeling strategies [Lee and others, 2013; Xie *et al.*, 2020b] (see Fig. 2, (c) left). However, directly applying these methods to the SSMAR task presents

*Corresponding authors.

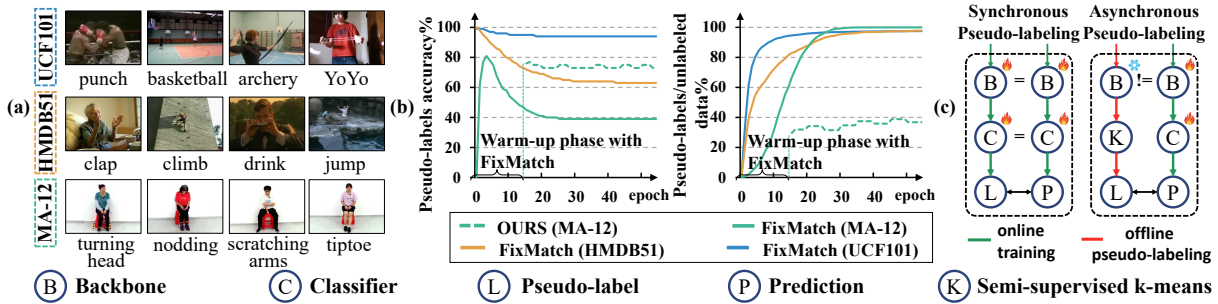


Figure 2: (a) Micro-actions (MA-12) are less distinct from each other and more difficult to differentiate than conventional actions (UCF101 and HMDB51). (b) Comparison of training process between our method and Fixmatch on the traditional action recognition datasets and a MAR dataset with 50% labeled data. FixMatch performs well on traditional action recognition datasets (HMDB51 and UCF101). However, when applying it on a MAR dataset (MA-12), as training proceeds, the number of unlabeled samples that pass the set threshold for the online pseudo-labeling method gradually increases, and the accuracy of the pseudo-labeling gradually decreases. In contrast, our method consistently generates high-accurate pseudo-labels. (c) FixMatch performs a synchronous pseudo-labeling and training. Instead, the proposed asynchronous approach separates the pseudo-labeling from the training process, where the pseudo-labels are first obtained by semi-supervised clustering in the offline phase and are then utilized for the online model training.

a new challenge, *i.e.*, the model tends to accumulate pseudo-labeling errors more significantly in the later stages of training. As shown in Fig. 2 (b), on a traditional action recognition dataset, UCF101 [Soomro and others, 2012], the popular method FixMatch [Sohn *et al.*, 2020] is effective in mining a large number of accurate pseudo-labels. However, as the complexity of the action recognition task increases, such as HMDB51 [Kuehne *et al.*, 2011], the accuracy of pseudo-labels declines significantly and the issue of overfitting to incorrect pseudo-labels becomes increasingly severe. The problem is particularly acute in the context of MAR dataset, *e.g.*, MA-12 [Guo *et al.*, 2024] (examples of different tasks are shown in Fig. 2 (a)). Specifically, when applying FixMatch to the MA-12 dataset, the accuracy of pseudo-labels drops sharply from an initially high level (80%) after early training stages. Meanwhile, the model grows increasingly confident in its predictions and assigns pseudo-labels to all samples. These two observations indicate that FixMatch leads to an increasing number of incorrect pseudo-labels in late training stages. We argue that the primary cause of the overfitting issue is that the model relies on its own predictions as pseudo-labels while simultaneously using those same pseudo-labels as supervisory signals for training. This approach works well when the task is relatively simple, as shown with UCF101, where it generates a large number of accurate pseudo-labels. However, as task complexity grows, this approach becomes more prone to inaccurate pseudo-labels, leading to error accumulation and performance degradation, as observed on the MAR dataset.

To tackle this challenge, we introduce a novel framework, called Asynchronous Pseudo Labeling and Training (APLT), which explicitly decouples the pseudo-labeling from the model training process (see Fig. 2, (c) right). The framework operates in two phases: 1) offline pseudo-labeling that aims to generate accurate pseudo-labels through a semi-supervised clustering approach, and 2) online training that focuses on training the model using a prototype classifier in a robust manner. *In the offline pseudo-labeling phase*, we propose a non-parametric, semi-supervised k-means clustering

algorithm to generate accurate pseudo-labels. Specifically, we use labeled data as anchor points to cluster unlabeled data, thereby generating initial pseudo-labels. To enhance stability, we introduce a labeled-augmentation technique that increases the diversity of labeled samples, making the clustering anchors more robust for guiding the clustering process. In addition, a self-adaptive thresholding strategy is proposed to filter out less reliable pseudo-labels. We then build a memory-based prototype classifier by averaging the features of samples that are assigned with pseudo-labels for each cluster. *In the online training phase*, the pseudo-labels generated in the offline phase are used to supervise the outputs of the memory-based prototype classifier. The prototype classifier remains fixed during training and is updated only in the offline phase. The offline and online phases are performed alternately, helping the model to avoid the overfitting problem in an asynchronous manner. Moreover, we reduce the update frequency of pseudo-labeling to several epochs instead of one epoch, further mitigating the overfitting risk. As shown in Fig. 2 (b), our method is more resistant to overfitting on inaccurate pseudo-labels compared to FixMatch.

The main contributions are summarized as follows:

- We introduce a new setting, SSMAR, designed to reduce the annotation requirements of MAR. Additionally, we identify a critical challenge in applying the FixMatch to SSMAR, *i.e.*, overfitting on incorrect pseudo-labels.
- We propose a novel framework for SSMAR that explicitly separates the pseudo-labeling process from model training, making them asynchronous. This strategy can effectively mitigate the overfitting issue.
- Within our framework, we propose several strategies to enhance the reliability of pseudo-labels during the offline pseudo-labeling phase. Furthermore, we develop a memory-based prototype classifier to mitigate the overfitting issue during the online model training phase.
- Experiments on three SSMAR benchmarks demonstrate that our method significantly improves the performance of FixMatch, achieving state-of-the-art results.

2 Related Works

Micro-Action Recognition (MAR). Unlike the traditional action recognition, MAR focuses on classifying subtle and transient human body movements. To facilitate the study of MAR, multiple MAR datasets are proposed. The iMiGUE dataset [Liu *et al.*, 2021] combines fine-grained gesture actions with sentiment labels for micro-gesture understanding and sentiment analysis. Similarly, Chen *et al.* [Chen *et al.*, 2023b] propose the SMG dataset, where participants perform various micro-gestures by narrating both false and true stories. Building on this, the MA-52 dataset [Guo *et al.*, 2024] is proposed by capturing natural micro-actions from psychological interviews. Traditional action recognition has been benefited from large-scale labeled datasets to achieve high accuracy [Caba Heilbron *et al.*, 2015; Kay *et al.*, 2017; Soomro and others, 2012], allowing models [Wang *et al.*, 2018; Lin and others, 2019; Bertasius and others, 2021]. However, obtaining labeled MAR datasets is much more difficult. *The proposed SSMAR setting aims to utilize both labeled and unlabeled data to train a MAR model that can be comparable with the one trained with fully-labeled data.*

Semi-Supervised Learning (SSL). SSL has gained significant attention in the community, aiming to leverage abundant unlabeled data alongside limited labeled data to enhance model performance. Two prominent SSL trends are pseudo-labeling and consistency regularization. Pseudo-labeling approaches [Lee and others, 2013; Xie *et al.*, 2020b] rely on adding high-confidence pseudo-labels to the training dataset. Consistency regularization approaches [Rasmus *et al.*, 2015; Ke *et al.*, 2019; Oliver *et al.*, 2018] assume that applying perturbations to input samples or features does not change the outputs of the model. FixMatch [Sohn *et al.*, 2020] combines pseudo-labeling and consistency regularization by using high-confidence pseudo-labels of weakly augmented samples to supervise strongly augmented samples. SoftMatch [Chen *et al.*, 2023a] uses a Gaussian function to assign weights to samples, resolving the trade-off between quantity and quality of pseudo-labels. *The above methods mainly are designed for image classification task. Instead, we propose a novel SSL framework for the MAR task, which performs pseudo-labeling and model training in an asynchronous way.*

Semi-Supervised Action Recognition (SSAR). The exploration of SSL in video recognition lags behind the progress in image classification. VideoSSL [Jing *et al.*, 2021] compares SSL methods that are specifically applied to videos, revealing limitations in extending pseudo-labeling directly. LTG [Xiao *et al.*, 2022] introduces temporal gradient as an additional modality to generate high-quality pseudo-labels for training. Recently, self-supervised learning has proven to be effective in learning powerful video representations [Dave *et al.*, 2022; Feichtenhofer *et al.*, 2021; Qian *et al.*, 2024]. TimeBalance [Dave *et al.*, 2023] leverages the temporal contrastive losses from TCLR [Dave *et al.*, 2022] to learn the temporal distinctive teacher. SVFormer [Xing *et al.*, 2023] explores the potential benefit of Video Transformers for SSAR. FinePseudo [Dave *et al.*, 2025] is proposed to improve pseudo-labeling through temporal alignment for fine-grained action recognition under SSL context. *Different from them,*

this work focuses on the SSMAR task, which is more difficult than traditional and fine-grained action recognition tasks.

3 Method

Task Definition. In SSMAR, we are given a set of human micro-action videos, defined as $\mathcal{D} = \{\mathcal{D}_l, \mathcal{D}_u\}$, where samples of \mathcal{D}_l are labeled while samples of \mathcal{D}_u are unlabeled. $\mathcal{D}_l = \{V_l^i, y_l^i\}_{i=1}^{N_l}$ consists of N_l videos and corresponding labels. The videos are from C categories, *i.e.*, the label y_l^i is derived from the set of labels $Y = \{1, 2, \dots, C\}$. $\mathcal{D}_u = \{U^i\}_{i=1}^{N_u}$ consists of N_u unlabeled videos. The goal of SSMAR is to leverage both labeled and unlabeled data to learn an effective MAR model.

3.1 Overview

As shown in Fig. 3, we propose an asynchronous pseudo-labeling and training framework, dubbed APLT, for SSMAR, which includes two phases: offline pseudo-labeling and online model training. *During offline phase*, we propose a non-parametric, semi-supervised k-means clustering method to obtain accurate pseudo-labels. In particular, labeled data serve as anchors to guide the clustering, producing initial pseudo-labels for unlabeled data. To improve robustness, we employ a labeled-augmentation approach that enhances labeled sample diversity, making the clustering anchors more dependable for guiding the clustering process. Additionally, a self-adaptive thresholding mechanism is presented to filter out less reliable pseudo-labels. Then, we construct a memory-based prototype classifier by averaging the features of samples assigned with the same cluster. *During online phase*, pseudo-labels produced in the offline phase are used to supervise the memory-based prototype classifier’s outputs. This classifier remains unchanged throughout training and is only updated in the offline phase. By alternating between offline and online phases, the model effectively avoids overfitting in an asynchronous manner. Note that, the FixMatch is also applied in our framework in default, which uses the predictions of the parametric classifier as pseudo-labels to supervise the same classifier.

3.2 Warm-Up

Our asynchronous learning strategy is implemented after a warm up stage with FixMatch. Specifically, given a labeled video V_l^i , we calculate the cross-entropy loss for labeled set by:

$$\mathcal{L}_{sup}^{logits} = \frac{1}{B} \sum_{i=1}^B \mathcal{H}(y_l^i, p_m(y | \omega(V_l^i))), \quad (1)$$

where B is the batch size, $\omega(\cdot)$ indicates the weak data augmentation function, $p_m(\cdot)$ is the output probability from the parametric classifier, y_l^i is the ground-truth label for V_l^i , and $\mathcal{H}(\cdot, \cdot)$ is the cross-entropy loss.

For the unlabeled set, we use a similar loss function but generate pseudo-labels for unlabeled data as we do not have ground-truth labels for them. The loss for unlabeled data is formulated as:

$$\mathcal{L}_u^{logits} = \frac{1}{B} \sum_{i=1}^B \mathbf{1}(\max(q_i) \geq \tau) \mathcal{H}(\hat{q}_i, p_m(y | \mathcal{A}(U^i))), \quad (2)$$

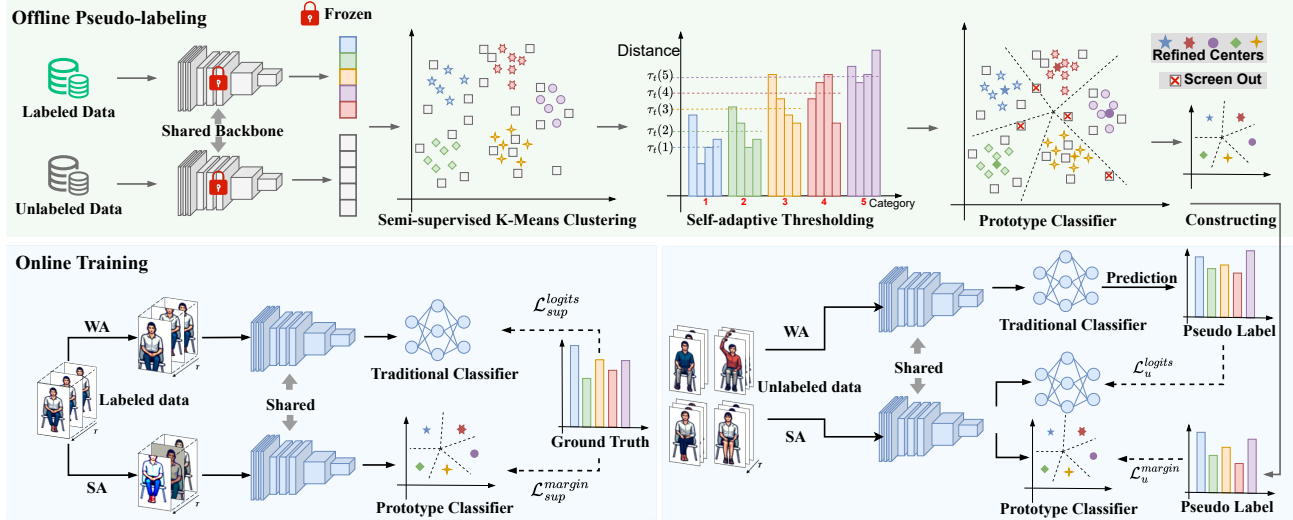


Figure 3: Overview of the proposed APLT framework. APLT includes two phases: offline pseudo-labeling and online model training. During the offline phase, we propose an approach to generate reliable pseudo-labels by semi-supervised clustering and self-adaptive thresholding. In addition, we construct a memory-based prototype classifier by averaging features assigned with the same cluster. During the online phase, we augment samples for both labeled and unlabeled samples. For the labeled data, we use the ground-truth labels to supervise the two classifiers ($\mathcal{L}_{sup}^{margin}$ and $\mathcal{L}_{sup}^{logits}$). For the unlabeled data, we use the predictions of traditional classifier to supervise the same classifier (\mathcal{L}_u^{logits}) while use the pseudo-labels generated by the offline phase to supervise the prototype classifier (\mathcal{L}_u^{margin}). “WA” and “SA” stand for weak augmentation and strong augmentation, respectively.

where q_i is the predicted class distribution of unlabeled video data after weak augmentation: $q_i = p_m(y | \omega(U^i))$. $\hat{q}_i = \arg \max(q_i)$ is the predicted pseudo-label. $\mathcal{A}(\cdot)$ means the strong data augmentation function. $\mathbf{1}(\cdot)$ is an indicator function of the confidence-based threshold and τ is the threshold.

Overall, the loss function for warm up stage is:

$$\mathcal{L}^{logits} = \mathcal{L}_{sup}^{logits} + \mathcal{L}_u^{logits}. \quad (3)$$

After training the model with the basic FixMatch for several epochs, we will additionally include the proposed asynchronous method, which includes the offline pseudo-labeling and online model training phases.

3.3 Phase I: Offline Pseudo-Labeling

To mitigate the risk of overfitting on incorrect supervisions during training, we propose to generate pseudo-labels by a non-parametric clustering method instead of directly using the output of the parametric classifier, which includes three strategies: semi-supervised clustering, labeled-augmentation, and self-adaptive threshold strategy.

Semi-supervised Clustering. The key of semi-supervised clustering is using labeled data as anchors to guide the clustering on unlabeled data, which is inspired by [Vaze *et al.*, 2022]. Specifically, after a training epoch, we first generate features for all data, $F = \{F_u, F_l\}$, in which the feature is obtained by the feature extractor f_θ . F_l and F_u represent the features of labeled and unlabeled data respectively. We then initialize the C centroids by averaging the features of labeled data for each class and implement the following two stages. 1) Cluster Assignment: Each unlabeled instance is assigned with a cluster label by identifying its nearest centroid. 2) Center Update: The centroids are updated by averaging

all data features within each cluster, where the cluster labels are ground-truth labels for labeled data and are the pseudo-labels generated by first stage (cluster assignment) for unlabeled data, respectively. We iteratively repeat the above two stages until the algorithm converges, resulting in clustering pseudo-labels for unlabeled data.

Labeled-Augmentation. To improve the clustering stability, we introduce the labeled-augmentation by applying strong augmentation to the labeled data during the semi-supervised clustering. In this way, we can obtain the augmented labeled set \mathcal{D}_{sl} which is also utilized to initialize and update centers in the clustering process. This strategy enables us obtain more robust clustering centers and thus produce more accurate pseudo-labels for unlabeled data.

Self-Adaptive Threshold Strategy. During clustering, hard samples will be assigned with wrong clustering labels. Using such incorrect pseudo-labels for model training will hamper the optimization. Thus, it is important to select reliable pseudo-labels. One common solution is using a threshold to filter out pseudo-labels with low-confidence. However, since micro-action categories are diverse, the difficulties of recognizing them are very different. Thus, using a single fixed threshold to constrain all classes is not reasonable. For example, the average confidence level for the head movement-related category is typically higher than that for the upper limb movement-related category, as recognizing the former category is more easy.

To solve this problem, we propose a self-adaptive thresholding strategy, which is calculated by both global threshold and category-specific local thresholds. Specifically, given an unlabeled sample, we use its distance from the assigned centroid as the confidence indicator. The global threshold τ_{global}

is set as the average distance of unlabeled data from their corresponding assigned clusters, reflecting the overall clustering status. Local thresholds respond to the clustering state of each class, in which we compute the class-specific local threshold for each cluster. By considering both global and local thresholds, our self-adaptive thresholding can be expressed as:

$$\begin{cases} \tau_{global} = \frac{1}{N_u} \sum_{i=1}^{N_u} dis(U^i), \\ \tau_{local}(c) = \frac{1}{N_u^c} \sum_{i=1}^{N_u^c} dis(U_c^i), \\ \tau_{adapt}(c) = \frac{\tau_{local}(c)}{\max(\tau_{local})} \cdot \tau_{global}, \end{cases} \quad (4)$$

where U_c^i represents an instance assigned with category c in \mathcal{D}_u and N_u^c is the number of unlabeled instances assigned with category c . $dis(\cdot)$ indicates the distance to the assigned class center for an unlabeled instance. $\tau_{adapt}(c)$ is the adaptive threshold of class c . $\tau_{local} = [\tau_{local}(1), \tau_{local}(2), \dots, \tau_{local}(C)]$ is the set of local thresholds. Given an unlabeled sample U^i , if its distance to the assigned center c is lower than $\tau_{adapt}(c)$, we regard its predicted cluster label as a reliable pseudo label. Otherwise, we ignore its pseudo label. By doing so, we could obtain a filtered pseudo-label set $\mathcal{D}_{su} = \{U_p^i, y_p^i\}_{i=1}^{N_p}$ with N_p instances. **Memory-based Prototype Classifier.** Most of the previous SSL approaches rely on the supervision of the output of the parametric-classifier during training on unlabeled data. Instead, we build a non-parametric memory-based prototype classifier based on the features of all data. Specifically, for a class c , we average all features of labeled data with class label c and unlabeled data assigned with pseudo-label c , obtaining corresponding prototype feature:

$$\rho_c = \frac{1}{N_c} \sum_i^{N_c} F_c^i, \quad (5)$$

where N_c is the number of samples belonging to class c . F_c^i is the feature of an instance belonging to class c . Thus, the prototype feature is $\rho = \{\rho_1, \rho_2, \dots, \rho_C\}$, constructing the memory-based prototype classifier. This non-parametric classifier is directly added after the backbone for model training, like the parametric one.

3.4 Phase II: Online Training

During the online training, we not only train the model with the basic FixMatch loss based on parametric classifier but also with a margin loss based on the non-parametric classifier. Specifically, given a sample with strong data augmentation, we first calculate its feature and obtain the prediction based on the non-parametric classifier. The margin loss for labeled data is formulated as:

$$\mathcal{L}_{sup}^{margin} = -\frac{1}{B} \sum_{i=1}^B \log \frac{\exp(F_l^i \cdot \rho_c)}{\sum_{i=1}^C \exp(F_l^i \cdot \rho_i)}, \quad (6)$$

where F_l^i is the feature of the labeled sample i obtained by the current training model and ρ_c is the prototype feature corresponding to the ground-truth c .

Similarly, the margin loss for unlabeled data is as:

$$\mathcal{L}_u^{margin} = -\frac{1}{B} \sum_{i=1}^B \log \frac{\exp(F_u^i \cdot \rho_c)}{\sum_{i=1}^C \exp(F_u^i \cdot \rho_i)}, \quad (7)$$

where F_u^i is the feature of the unlabeled sample i and ρ_c is the prototype feature of the assigned pseudo-label c of the unlabeled sample i .

The final margin loss can be formulated as:

$$\mathcal{L}^{margin} = \mathcal{L}_{sup}^{margin} + \mathcal{L}_u^{margin}. \quad (8)$$

By combining with the basic Fixmatch, our final loss is defined as:

$$\mathcal{L} = \mathcal{L}^{logit} + \lambda \mathcal{L}^{margin}, \quad (9)$$

where λ is a hyperparameter that balances the contribution between \mathcal{L}^{logit} and \mathcal{L}^{margin} .

3.5 Phase Iteration

In our framework, the offline clustering phase and the online model training phase are performed alternately. Specifically, during model training, the memory-based classifier remains fixed and is used only to produce predictions. During the offline clustering, we both recalculate pseudo-labels based on the most recent model and update the prototypes of the memory-based classifier. Unlike FixMatch, this approach avoids creating pseudo-labels, updating the memory-based classifier, and producing predictions in the same iteration and manner, thereby mitigating overfitting to incorrect pseudo-labels. Furthermore, because deep networks possess a strong capacity to overfit, updating the prototypes and pseudo-labels too frequently can also lead to overfitting issue. Consequently, we run the online phase every 10 training epochs instead of every epoch.

During testing, we discard the parametric classifier and use the outputs of the memory-based classifier as the final predictions.

4 Experiments

4.1 Dataset and Experimental Setup

Benchmarks: We build semi-supervised micro-action recognition datasets based on three datasets, *i.e.*, MA-52 [Guo *et al.*, 2024], iMiGUE [Liu *et al.*, 2021], and SMG [Chen *et al.*, 2023b]. *Note that, many classes in these three datasets are not satisfied for semi-supervised learning, as they do not have insufficient samples.* For example, in the SMG dataset, actions like *Touching or covering* and *Pulling shirt collar* have only 11 instances each. Hence, we only select classes with sufficient samples to form the datasets, results in MA-12, iMiGUE-11, and SMG-5, each of which has 12, 11, and 5 classes respectively.

Experimental Setup. For each video, we use 8-frame clips for training and testing. We use the ResNet-18 [He *et al.*, 2016] as the backbone in default. For the warm up stage, we train the model with 15 epochs with only the FixMatch loss function. We then introduce our asynchronous learning strategy and train the model with additional 40 epochs. The model is updated by SGD optimizer with a momentum of 0.9 and a weight decay of 0.0005. The learning rate is set to 0.002 and follows a cosine decay schedule. The offline clustering phase is implemented after every 10 training epochs.

Method	MA-12								SMG-5	iMiGUE-11
	Resnet-18				Resnet-50				Resnet-18	Resnet-18
	10%	25%	40%	50%	10%	25%	40%	50%	20%	30%
Baseline (Labeled Only)	24.8	30.7	35.8	36.3	23.8	31.7	34.8	38.0	51.6	38.0
Pseudolabel [Lee and others, 2013]	26.9	32.7	34.6	35.6	17.6	29.8	40.1	39.3	57.2	32.3
Mean Teacher [Tarvainen and Valpola, 2017]	21.5	26.0	31.0	14.4	20.0	23.6	37.5	34.5	48.0	30.9
VAT [Miyato <i>et al.</i> , 2018]	22.9	23.8	13.7	25.6	21.7	8.5	23.1	29.8	50.4	26.7
MixMatch [Berthelot <i>et al.</i> , 2019]	21.7	25.9	23.4	23.1	19.1	35.7	43.0	42.5	42.4	34.4
UDA [Xie <i>et al.</i> , 2020a]	27.7	32.8	38.3	41.3	25.2	37.0	41.6	43.3	52.8	40.0
FixMatch [Sohn <i>et al.</i> , 2020]	28.0	35.5	38.8	42.8	27.8	35.8	44.5	44.2	56.0	41.6
FlexMatch [Zhang <i>et al.</i> , 2021]	25.6	31.8	37.8	44.5	25.3	33.3	42.6	43.7	56.8	41.6
FreeMatch [Wang <i>et al.</i> , 2022]	26.7	35.4	39.7	43.1	25.8	33.3	42.1	43.2	53.6	40.7
SoftMatch [Chen <i>et al.</i> , 2023a]	24.6	28.3	34.3	34.6	26.8	32.4	37.5	36.1	57.2	36.4
InfoMatch [Han <i>et al.</i> , 2024]	24.4	32.4	35.3	41.0	22.8	31.3	36.7	36.4	55.2	41.5
FineSSL [Gan and Wei, 2024]	27.9	30.9	32.3	33.3	28.1	31.1	31.9	33.1	57.6	32.4
APLT (Ours)	34.1	43.8	52.9	57.3	34.7	50.5	57.8	63.9	59.2	43.8

Table 1: Performance (%) comparison with traditional SSL methods. Results are evaluated on MA-12 (↑), SMG-5 (↑) and iMiGUE-11 (↑).

4.2 Comparison with State of The Arts

Comparison with Traditional SSL Methods. We first compare our method with 11 state-of-the-art SSL approaches designed for image classification. For a fair comparison, we use the same backbone and experimental settings (e.g., data augmentation, optimizer, and training epochs) for all methods. The results on MA-12, SMG-5, and iMiGUE-11 are shown in Table 1. Our method clearly outperforms all baselines in every setting. For instance, with only 10% labeled data on MA-12, APLT surpasses FixMatch [Sohn *et al.*, 2020] by 6.1%. Moreover, most of the compared methods fail to improve on MA-12 with 10% and 25% labeled data, reflecting the challenges posed by SSMAR. In contrast, our method significantly boosts the baseline under both conditions. We further demonstrate that APLT benefits from a more powerful backbone. For example, using ResNet-50 instead of ResNet-18 yields an additional 6.6% improvement on MA-12 with 50% labeled data. However, most of the other methods show only marginal or even negative gains when switching to ResNet-50.

Comparison with SSL Methods for Action Recognition. We also compare with the art SSL methods designed for action recognition, including TCL [Singh *et al.*, 2021], LTG [Xiao *et al.*, 2022] and SVFormer [Xing *et al.*, 2023]. Results on MA-12 are shown in Table 2. We can observe that our approach consistently achieves higher accuracies over all settings over the compared methods in which LTG and SVFormer use more complex backbones (3D-ResNet-18 or ViT-B). This further demonstrates the superiority of our method in SSMAR over previous SSL methods.

Further Comparison with FixMatch. Fig. 4 (left) illustrates the class-specific gains of APLT compared to FixMatch [Sohn *et al.*, 2020]. We can find that the majority of classes benefit from our method and only two classes show comparable performance to FixMatch. Fig. 4 (right) provides a qualitative comparison between our APLT and FixMatch. Our approach accurately identifies different micro-actions that are misclassified by FixMatch.

Method	Backbone	Labeled Ratio			
		10%	25%	40%	50%
Baseline (Labeled Only)	ResNet-18	24.8	30.7	35.8	36.3
TCL[Singh <i>et al.</i> , 2021]	ResNet-18	16.3	25.0	32.1	37.3
LTG [Xiao <i>et al.</i> , 2022]	3D-ResNet-18	9.5	18.7	32.9	34.9
SVFormer [Xing <i>et al.</i> , 2023]	ViT-B	32.3	37.3	38.3	44.9
APLT (Ours)	ResNet-18	34.1	43.8	52.9	57.3

Table 2: Comparisons with state-of-the-art SSL methods for action recognition methods on MA-12 (↑).

4.3 Ablation Study

Analysis of Components of Non-Parametric Clustering. In Table 3, we evaluate the effectiveness of the components of the proposed non-parametric clustering. Based on the FixMatch (+SSL), we further train the model with variants of our method, in which we generate pseudo-labels by different methods, including pure k-means (+KM), semi-supervised k-means (+SSKM), and SSKM with labeled-augmentation (+LA) and self-adaptive thresholding (+SAT). We also evaluate the effect of training the model with weak or strong augmentation. We make the following conclusions. First, semi-supervised learning obtains limited improvement for SSMAR. Second, using non-parametric clustering to generate pseudo-labels can significantly improve the accuracy. Third, updating the model with strong augmentations achieves slightly better results. Fourth, all the proposed components can consistently increase the performance and our full method obtains the best results on all settings.

Effectiveness of Non-Parametric Classifier and Asynchronous Strategy. The proposed non-parametric classifier and the asynchronous pseudo-labeling model training strategy are both critical components of our approach. In Table 4, we evaluate the contributions of these two techniques. Specifically, when applying the asynchronous strategy to FixMatch, we generate pseudo-labels via the parametric classifier in an offline manner and keep them fixed during training. Conversely, when using a synchronous strategy in our method, we take the non-parametric classifier’s online outputs as pseudo-labels, rather than the clustering results. We observe the fol-

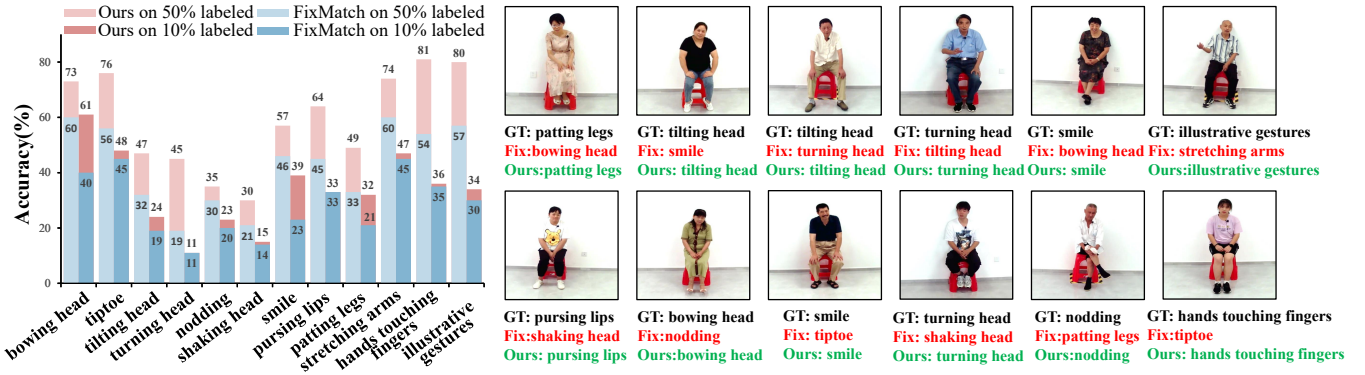


Figure 4: **Left:** Class accuracy comparison between APLT with FixMatch for MA-12 with 10% and 50% labeled data. **Right:** Visualization of the predictions of APLT and FixMatch. The two methods are trained with ResNet-18.

Method	Labeled Ratio		
	25%	40%	50%
Baseline (Labeled Only)	32.0	36.0	41.4
SSL	35.5	38.8	42.8
SSL + KM	39.8	45.9	50.1
SSL + SSKM (W)	40.2	48.0	51.7
SSL + SSKM (S)	41.0	50.7	53.8
SSL + SSKM (S) + LA	42.5	52.1	56.0
SSL + SSKM (S) + SAT	41.7	51.8	54.6
SSL + SSKM (S) + LA + SAT	43.8	52.9	57.3

Table 3: Ablation study on MA-12 (\uparrow). **SSL:** FixMatch, **KM:** K-Means, **SSKM:** Semi-Supervised K-Means, **W:** Weak augmentation, **S:** Strong augmentation, **LA:** Labeled-Augmentation, **SAT:** Self-Adaptive Thresholding.

MA-12 (50% labeled data)		
Method	Update Strategy	Top-1 Acc.
FixMatch (Baseline)	SYN	42.8
FixMatch	ASY	43.4
APLT (Ours)	SYN	54.3
APLT (Ours)	ASY	57.3

Table 4: Evaluation on synchronous strategy (SYN) and asynchronous strategy (ASY) on MA-12.

lowing: 1) Using our non-parametric classifier substantially improves accuracy over FixMatch (Baseline), even under the synchronous strategy; 2) Both FixMatch and our APLT benefit from the asynchronous strategy, with APLT achieving a higher performance gain. These findings validate the effectiveness of both the proposed non-parametric classifier and the asynchronous strategy. Furthermore, when pseudo-labels and classifier outputs are generated through different mechanisms (*e.g.*, our APLT), the asynchronous strategy delivers even stronger benefits.

Further Evaluation on Non-Parametric Classifier. To further evaluate the benefit of the non-parametric classifier, we evaluate two variants. 1) Variant I: We remove the non-parametric classifier but further use the pseudo-labels generated by the clustering to train the parametric classifier that is used for testing. 2) Variant II: We train the model with only

Method	Labeled Ratio		
	25%	40%	50%
Baseline	30.7	35.8	36.3
Variant I of APLT	39.3	44.0	50.8
Variant II of APLT	37.6	44.9	48.1
APLT (Ours)	43.8	52.9	57.3

Table 5: Effect of non-parametric classifier on MA-12. Variant I: Directly applying clustering labels on parametric classifier. Variant II: Only using non-parametric classifier during testing.

the basic FixMatch loss, *i.e.*, the model is trained without the loss of \mathcal{L}^{margin} . However, we implement clustering with the final model and build the non-parametric classifier for testing. Results in Table 5 show that without using the non-parametric classifier during training significantly reduces the accuracies. This indicates the importance of the non-parametric classifier in our approach. On the other hand, only using the non-parametric classifier can also achieve a clearly higher results than the baseline, further indicating the high-quality of the pseudo-labels generated by our non-parametric clustering.

5 Conclusion

In this work, we introduce the Semi-Supervised MAR (SS-MAR) setting. Through our evaluation, we show that traditional Semi-Supervised Learning (SSL) methods are prone to overfitting on inaccurate pseudo-labels. To overcome this, we present the Asynchronous Pseudo Labeling and Training (APLT) framework, which decouples the pseudo-labeling process from model training. In the offline pseudo-labeling phase, we propose a semi-supervised clustering approach to generate accurate pseudo-labels. During the online model training phase, we optimize the model with the proposed memory-based prototype classifier and the generated pseudo-labels. By alternating between pseudo-labeling and training phases asynchronously, APLT effectively mitigates error accumulation and enhances the accuracy. Extensive experiments on three MAR datasets validate the superiority of APLT, achieving significant performance gains over state-of-the-art SSL methods.

Acknowledgements

This work has been supported by the New Cornerstone Science Foundation through the XPLOER PRIZE. This work was also funded by the National Natural Science Foundation of China (No. 72188101 & No. 62302140 & No. 62472139) and the Fundamental Research Funds for the Central Universities (No. JZ2024HGTB0261 & No. JZ2024HGTA0178). The computation is completed on the HPC Platform of Hefei University of Technology.

References

- [Arnab *et al.*, 2021] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021.
- [Bertasius and others, 2021] Bertasius et al. Is space-time attention all you need for video understanding? In *Proceedings of International Conference on Machine Learning*, 2021.
- [Berthelot *et al.*, 2019] 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*, 2019.
- [Caba Heilbron *et al.*, 2015] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015.
- [Chen *et al.*, 2021] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021.
- [Chen *et al.*, 2023a] 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.
- [Chen *et al.*, 2023b] Haoyu Chen, Henglin Shi, Xin Liu, Xiaobai Li, and Guoying Zhao. Smg: A micro-gesture dataset towards spontaneous body gestures for emotional stress state analysis. *International Journal of Computer Vision*, 2023.
- [Dave *et al.*, 2022] Ishan Dave, Rohit Gupta, Mamshad Nayeem Rizve, and Mubarak Shah. Tclr: Temporal contrastive learning for video representation. *Computer Vision and Image Understanding*, 2022.
- [Dave *et al.*, 2023] Ishan Rajendrakumar Dave, Mamshad Nayeem Rizve, Chen Chen, and Mubarak Shah. Timebalance: Temporally-invariant and temporally-distinctive video representations for semi-supervised action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.
- [Dave *et al.*, 2025] Ishan Rajendrakumar Dave, Mamshad Nayeem Rizve, and Mubarak Shah. Finepseudo: improving pseudo-labelling through temporal-alignability for semi-supervised fine-grained action recognition. In *European Conference on Computer Vision*, 2025.
- [Feichtenhofer *et al.*, 2019] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019.
- [Feichtenhofer *et al.*, 2021] Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [Gan and Wei, 2024] Kai Gan and Tong Wei. Erasing the bias: Fine-tuning foundation models for semi-supervised learning. *arXiv preprint arXiv:2405.11756*, 2024.
- [Guo *et al.*, 2024] Dan Guo, Kun Li, Bin Hu, Yan Zhang, and Meng Wang. Benchmarking micro-action recognition: Dataset, method, and application. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.
- [Han *et al.*, 2024] Qi Han, Zhibo Tian, Chengwei Xia, and Kun Zhan. Infomatch: Entropy neural estimation for semi-supervised image classification. *arXiv preprint arXiv:2404.11003*, 2024.
- [He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.
- [Jing *et al.*, 2021] Longlong Jing, Toufiq Parag, Zhe Wu, Yingli Tian, and Hongcheng Wang. Videoss1: Semi-supervised learning for video classification. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021.
- [Kay *et al.*, 2017] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.
- [Ke *et al.*, 2019] Zhanghan Ke, Daoye Wang, Qing Yan, Jimmy Ren, and Rynson W.H. Lau. Dual student: Breaking the limits of the teacher in semi-supervised learning. *Cornell University - arXiv, Cornell University - arXiv*, 2019.
- [Kuehne *et al.*, 2011] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 2011.
- [Lee and others, 2013] 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*, 2013.
- [Lin and others, 2019] Lin et al. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of*

- the IEEE International Conference on Computer Vision*, 2019.
- [Liu *et al.*, 2021] Xin Liu, Henglin Shi, Haoyu Chen, Zitong Yu, Xiaobai Li, and Guoying Zhao. imigue: An identity-free video dataset for micro-gesture understanding and emotion analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021.
- [Luo and Yang, 2020] Wenfeng Luo and Meng Yang. Semi-supervised semantic segmentation via strong-weak dual-branch network. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, 2020.
- [Miyato *et al.*, 2018] 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*, 2018.
- [Noroozi *et al.*, 2021] Fatemeh Noroozi, Ciprian Adrian Corneanu, Dorota Kaminska, Tomasz Sapinski, Sergio Escalera, and Gholamreza Anbarjafari. Survey on emotional body gesture recognition. *IEEE Transactions on Affective Computing*, 2021.
- [Oliver *et al.*, 2018] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. *Advances in neural information processing systems*, 2018.
- [Qian *et al.*, 2024] Rui Qian, Weiyao Lin, John See, and Dian Li. Controllable augmentations for video representation learning. *Visual Intelligence*, 2024.
- [Rasmus *et al.*, 2015] Antti Rasmus, Harri Valpola, Mikko Honkala, Mathias Berglund, and Tapani Raiko. Semi-supervised learning with ladder networks. *Neural Information Processing Systems, Neural Information Processing Systems*, 2015.
- [Singh *et al.*, 2021] Ankit Singh, Omprakash Chakraborty, Ashutosh Varshney, Rameswar Panda, Rogerio Feris, Kate Saenko, and Abir Das. Semi-supervised action recognition with temporal contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.
- [Sohn *et al.*, 2020] 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*, 2020.
- [Soomro and others, 2012] Soomro *et al.* Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.
- [Tarvainen and Valpola, 2017] 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*, 2017.
- [Vaze *et al.*, 2022] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.
- [Wang *et al.*, 2018] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.
- [Wang *et al.*, 2022] 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.
- [Wang *et al.*, 2024] Fei Wang, Dan Guo, Kun Li, Zhun Zhong, and Meng Wang. Frequency decoupling for motion magnification via multi-level isomorphic architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.
- [Wu *et al.*, 2023] Jianlong Wu, Wei Sun, Tian Gan, Ning Ding, Feijun Jiang, Jialie Shen, and Liqiang Nie. Neighbor-guided consistent and contrastive learning for semi-supervised action recognition. *IEEE Transactions on Image Processing*, 2023.
- [Xiao *et al.*, 2022] Junfei Xiao, Longlong Jing, Lin Zhang, Ju He, Qi She, Zongwei Zhou, Alan Yuille, and Yingwei Li. Learning from temporal gradient for semi-supervised action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.
- [Xie *et al.*, 2018] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European conference on computer vision (ECCV)*, 2018.
- [Xie *et al.*, 2020a] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in neural information processing systems*, 2020.
- [Xie *et al.*, 2020b] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.
- [Xing *et al.*, 2023] Zhen Xing, Qi Dai, Han Hu, Jingjing Chen, Zuxuan Wu, and Yu-Gang Jiang. Svformer: Semi-supervised video transformer for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.
- [Zhang *et al.*, 2021] 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*, 2021.