ActiveDC: Distribution Calibration for Active Finetuning

Wenshuai Xu[1, 2], Zhenghui Hu [*2], Yu Lu[2], Jinzhou Meng[1, 2], Qingjie Liu[2, 3], Yunhong Wang[2, 3]

[1]School of Software, Beihang University
[2]Hangzhou Innovation Institute, Beihang University
[3]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University

Abstract

The pretraining-finetuning paradigm has gained popularity in various computer vision tasks. In this paradigm, the emergence of active finetuning arises due to the abundance of large-scale data and costly annotation requirements. Active finetuning involves selecting a subset of data from an unlabeled pool for annotation, facilitating subsequent finetuning. However, the use of a limited number of training samples can lead to a biased distribution, potentially resulting in model overfitting. In this paper, we propose a new method called ActiveDC for the active finetuning tasks. Firstly, we select samples for annotation by optimizing the distribution similarity between the subset to be selected and the entire unlabeled pool in continuous space. Secondly, we calibrate the distribution of the selected samples by exploiting implicit category information in the unlabeled pool. The feature visualization provides an intuitive sense of the effectiveness of our method to distribution calibration. We conducted extensive experiments on three image classification datasets with different sampling ratios. The results indicate that ActiveDC consistently outperforms the baseline performance in all image classification tasks. The improvement is particularly significant when the sampling ratio is low, with performance gains of up to 10%. Our code will be publicly available.

1. Introduction

The recent successes in deep learning owe much of their progress to the availability of extensive training data. However, it is crucial to recognize that annotating large-scale datasets demands a significant allocation of human resources. In response to this challenge, a prevalent approach has emerged, referred to as the pretraining-finetuning paradigm. This paradigm involves the initial pretraining of models on a substantial volume of data in an unsupervised fashion, followed by a subsequent finetuning phase on a more limited, labeled subset of data.

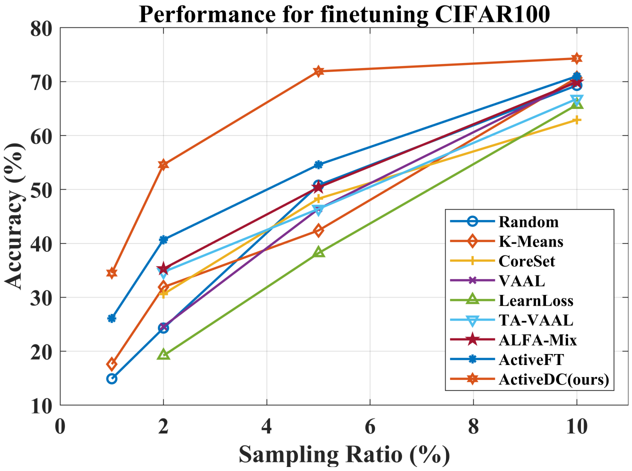

Figure 1. Comparison of the performance for finetuning the CIFAR100 dataset at different sampling ratios.

The existing body of literature extensively delves into the domains of unsupervised pretraining [9, 17, 46] and supervised finetuning [25, 42], making noteworthy contributions in these areas. However, when confronted with a vast repository of unlabeled data, the crucial task at hand is the judicious selection of the most valuable samples for annotation, a task necessitated by the constraints of limited annotation resources. At the same time, the distribution of the small number of labeled samples tends to significantly deviate from the overall distribution, raising the issue of how to calibrate the distribution of the selected samples [13, 43].

Active learning [2, 5, 22, 26, 28, 31, 37] is a method that iteratively selects the most informative samples for manual labeling during the training process, aiming to improve predictive model performance. Although active learning is considered a promising approach, empirical experiments

[*]Corresponding author

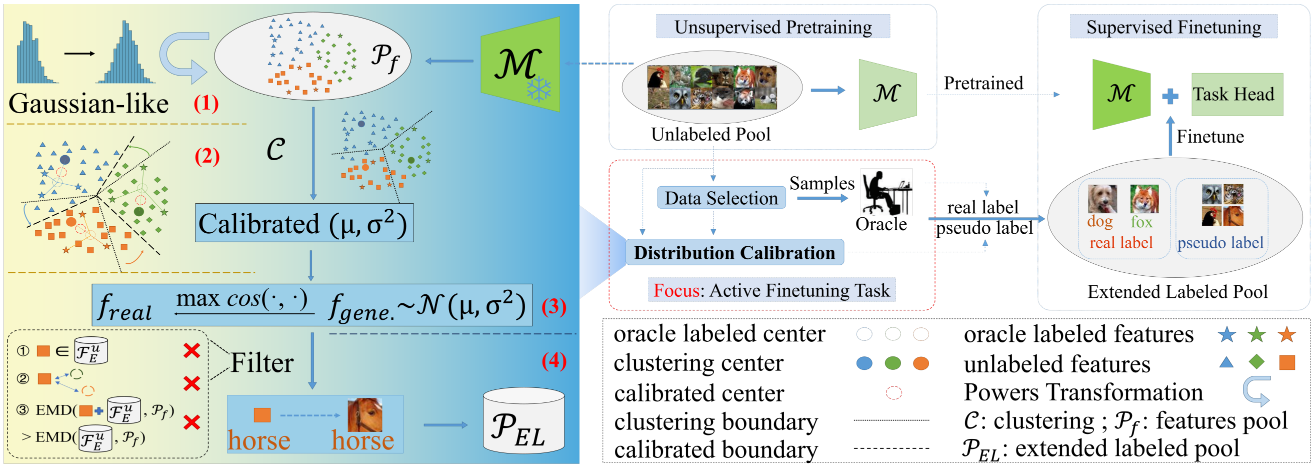

Figure 2. The active finetuning task involves the active selection of training data for finetuning within the pretraining-finetuning paradigm. We focus on data selection and distribution calibration from a large unlabeled data pool for annotation. The Distributed Calibration Module comprises four main steps: (1) applying Tukey's Ladder of Powers Transformation to render the feature distribution more Gaussian-like, (2) clustering the features and calibrating the statistics for different feature classes, (3) generating pseudo-features using the calibrated statistics and identifying the most similar real features, and (4) filtering and integrating the features into the extended labeled pool.

have uncovered its limitations [4, 16, 41] when employed within the context of the pretraining-finetuning paradigm. One plausible explanation for this phenomenon is the presence of more constrained annotation budgets and batch-selection strategy of active learning, which introduce biases into the training process, ultimately impeding its effectiveness.

In response to the limitations of traditional active learning within the pretraining-finetuning paradigm, a more efficient active finetuning approach, known as ActiveFT [41], has been developed to address these challenges. This method selects sample data by narrowing the distribution gap between the chosen subset and the entire unlabeled pool. Although the method exhibits promising results, its primary focus is distributional information. Unfortunately, it does not sufficiently harness information related to the number of known classification categories, and the implicit category-related data within a substantial volume of unlabeled pretrained features remains underutilized. Importantly, when working with a limited number of selected samples, there is *a heightened risk of bias* in how the chosen subset aligns with the overall distribution [13, 43]. Which, in turn, necessitates a larger sample size to rectify the distributional alignment.

In this paper, we present a novel method, **ActiveDC**, designed for enhancing active finetuning tasks. We introduce a novel distributional calibration technique to the active finetuning task. Our method leverages the comprehensive information derived from the entire feature data pool along with the labeled data, resulting in a significant improvement in model performance. Importantly, it achieves this enhancement while remaining cost-effective, eliminating the need for additional labeling efforts. Moreover, it does not incur excessive time consumption.

Specifically, our method comprises several key steps, as shown in Fig. 2. First, we employ a diversity selection strategy, such as ActiveFT, to select data for oracle annotation within limited budgets. Second, we proceed to normalize and transform the feature distribution of the entire data pool extracted by the pretrained model, followed by the application of clustering techniques. The resulting clustering categories are determined in reference to the labeled data, taking into account a trade-off between the center of clustering and the center of labeled samples, as dictated by the quantity of labeled samples available. Furthermore, we actively regulate the covariance of pseudo-categorical features. Subsequently, pseudo-features are generated based on the calibrated statistics, and real features most similar to these pseudo-features are identified through a similarity-based approach. The corresponding real data is then pseudo-labeled and integrated into the expanded labeled pool for finetuning. Throughout the iterative process of feature generation, certain filtering procedures are applied, contingent on the influence of the generated data on the extended labeled pool. Notably, the method excels in calibrating the distribution of a limited subset of training data selected by the active finetuning procedure, as depicted in Fig. 1.

Our main contributions are summarized as follows:

- We propose a new method, ActiveDC, aimed at calibrating data distributions for samples chosen through active finetuning techniques within limited labeling budgets. Our method significantly improves classification accuracy while maintaining the same budget.

- The Distributed Calibration Module exhibits inherent flexibility, allowing seamless integration with various active finetuning selection strategies.
- In the context of the classification task, our method yields significant performance improvements, with particular emphasis on its effectiveness at lower sampling ratios.

2. Related Work

Unsupervised Learning is designed to acquire feature representations without the need for labeled data. It plays a crucial role within the pretraining-finetuning paradigm. Besides, it exhibits notable effectiveness in mitigating the laborious task of data labeling [4]. Both contrastive methods [6, 9, 10, 15, 17] and generative methods [3, 18, 33, 38, 45] have achieved remarkable success in this field. The core idea behind contrastive learning is to train a model to distinguish between similar and dissimilar pairs of data samples. Building on the successes of existing contrastive learning techniques and the remarkable achievements of vision transformers in computer vision, innovative approaches like MoCov3 [11], DINO [7], and iBOT [46] have successfully extended the principles of contrastive learning to the realm of vision transformers [12], further enriching this vibrant field of study. Recent research in generative methods [3, 18] for predicting masked content within input samples has demonstrated promising performance over vision transformers. Extensive prior research [4, 16, 38, 41, 45] has thoroughly examined the advantageous contributions of both kinds of methods in the context of downstream supervised finetuning.

Active Learning aims at selecting the most valuable samples for labeling to optimize the model performance with a limited labeling budget. Much of the existing research in this field centers on sample selection strategies based on uncertainty [20, 21, 27, 34, 44] and diversity [2, 16, 31, 32] in pool-based scenarios. Uncertainty quantifies the model's level of perplexity when presented with data and can be estimated through various heuristics, including predictive probability, entropy, margin [20] and predictive loss [44]. Conversely, some algorithms seek to identify a subset that effectively represents the entire data pool by considering the diversity and representativeness of the data [8]. Diversity is quantified using measures such as Euclidean distance between global features [31], KL-divergence [1] between local representations, gradients spanning diverse directions [2], or adversarial loss [21, 32, 34], and more.

Active Finetuning is a task that actively selects training data for finetuning within the pretraining-finetuning paradigm. The majority of active learning algorithms mentioned above are primarily tailored for training models from scratch. However, prior research [4, 16, 41], has elucidated their adverse effects when applied to the finetuning process following unsupervised pretraining. Therefore, the introduction of the active finetuning task presents a novel approach to sample selection for labeling in a single pass [40, 41], especially in preparation for subsequent finetuning processes. An effective strategy, named ActiveFT [41], involves selecting data for labeling by converging the distribution of the chosen subset with that of the complete unlabeled pool within a continuous space. Nevertheless, unsupervised models, derived from extensive pretraining on large-scale datasets, exhibit robust feature extraction capabilities. As a result, within the pretraining-finetuning paradigm, the selection of samples often involves a limited quantity. In such cases, the inherent bias in the subset chosen by ActiveFT to align with the overall distribution tends to be relatively pronounced, necessitating a greater volume of samples to rectify the distribution effectively.

3. Methodology

This section delineates our novel active finetuning methodology. Section 3.1 initiates by furnishing a comprehensive overview of the pretraining-finetuning paradigm, followed by Sec. 3.2, which introduces the data selection module. Subsequently, Sec. 3.3 furnishes an exhaustive exposition of the distribution calibration module.

3.1. Overview

The complete pipeline for conducting the active finetuning task within the pretraining-finetuning paradigm is visually represented in Fig. 2. This paradigm consists of two distinct stages. In the initial stage, the model undergoes unsupervised pretraining on an extensive dataset, enabling it to traverse various classes of data features within the feature space. This stage establishes the foundation for subsequent feature extraction. In the next stage, the pretrained model is coupled with a task-specific module to facilitate supervised finetuning on a smaller, labeled subset tailored to specific tasks. The pivotal juncture between these two stages centers on the meticulous construction of the labeled subset. We select data for labeling based on fitting the distribution of the entire feature pool, and select appropriate data for pseudo-labeling by calibrating the category distributions to construct well-distributed training data for subsequent finetuning.

We formally define a deep neural network model $\mathcal{M}(\cdot;\omega_0) : \mathcal{X} \rightarrow \mathbb{F}^u$ with pretrained weight ω_0, where $\mathcal{X}$ is the data space and $\mathbb{F}^u$ is the normalized high dimensional feature space. We also have access to a large unlabeled data pool $\mathcal{P}^u = \{x_i\}_{i\in[N]} \sim p_u$ inside data space $\mathcal{X}$ with distribution p_u, where $[N] = \{1, 2, \cdots, N\}$. We design a sampling strategy $\mathcal{S} = \{s_j \in [N]\}_{j\in[B]}$ to select a subset $\mathcal{P}^u_{\mathcal{S}} = \{x_{s_j}\}_{j\in[B]} \subset \mathcal{P}^u$ from $\mathcal{P}^u$, where B is the annotation budget size for supervised finetuning. The model would have access to the labels $\{y_{s_j}\}_{j\in[B]} \subset \mathcal{Y}$ of this subset through the oracle, obtaining a labeled data pool $\mathcal{P}^l_{\mathcal{S}} = \{x_{s_j}, y_{s_j}\}_{j\in[B]}$, where $\mathcal{Y}$ is the label space. The normalized high dimensional feature pool $\mathcal{F}^u = \{f_i\}_{i\in[N]} \sim p_{f_u}$ has a distribution p_{f_u}. The feature pool $\mathcal{F}^u_{\mathcal{S}}$ is also associated with the selected data subset $\mathcal{P}^u_{\mathcal{S}}$, with the corresponding distribution over $\mathcal{F}^u_{\mathcal{S}}$ in the feature space denoted as $p_{f_{\mathcal{S}}}$.

3.2. Data Selection

The data selection strategy we employed is ActiveFT [41], guided by two basic intuitions: 1) bringing the distributions of the selected subset $\mathcal{P}^u_{\mathcal{S}}$ and the original pool $\mathcal{P}^u \sim p_u$ closer, and 2) maintaining the diversity of $\mathcal{P}^u_{\mathcal{S}}$. The first ensures the model finetuned on the subset performs similarly to one trained on the full set, while the second allows the subset to cover corner cases in the full set. The goal of distribution selection is to find the optimal selection strategy $\mathcal{S}$ as:

$$\mathcal{S}_{opt} = \arg\min_{\mathcal{S}} D(p_{f_u}, p_{f_{\mathcal{S}}}) - \lambda R(\mathcal{F}^u_{\mathcal{S}}) \tag{1}$$

where $D(\cdot,\cdot)$ is a distance metric between distributions, $R(\cdot)$ is used to assess the diversity of a set, and λ is a scaling factor to balance these two terms.

Optimizing the discrete selection strategy $\mathcal{S}$ directly is challenging. Therefore, it is better to model $p_{f_{\mathcal{S}}}$ with $p_{\theta_{\mathcal{S}}}$, where $\theta_{\mathcal{S}} = \{\theta^j_{\mathcal{S}}\}_{j\in[B]}$ are the continuous parameters and B is the annotation budget size. Each $\theta^j_{\mathcal{S}}$ after optimization corresponds to the feature of a selected sample $f_{\mathcal{S}_j}$. The feature $f_{\mathcal{S}_j}$ closest to each $\theta^j_{\mathcal{S}}$ can be found after optimization to determine the selection strategy $\mathcal{S}$. Therefore, the goal in Eq. (1) can be expressed as:

$$\theta_{\mathcal{S},opt} = \arg\min_{\theta_{\mathcal{S}}} D(p_{f_u}, p_{\theta_{\mathcal{S}}}) - \lambda R(\theta_{\mathcal{S}}) \;\; s.t. \;\; \left\|\theta^j_{\mathcal{S}}\right\|_2 = 1 \tag{2}$$

The protocol outlined in [7] is followed, and the cosine similarity between normalized features is utilized as the metric, denoted as $cos(f_1, f_2) = f_1^T f_2, \|f_1\|_2 = \|f_2\|_2 = 1$. For each $f_i \in \mathcal{F}^u$, there exists a $\theta^{c_i}_{\mathcal{S}}$ most similar (and closest) to f_i, i.e.

$$c_i = \arg\max_{j\in[B]} cos(f_i, \theta^j_{\mathcal{S}}) \tag{3}$$

where c_i is continuously updated in the optimization process.

Thus, the following loss function can be continuously optimized to address Eq. (2):

$$\begin{aligned} L &= \arg\min_{\theta_{\mathcal{S}}} D(p_{f_u}, p_{\theta_{\mathcal{S}}}) - \lambda \cdot R(\theta_{\mathcal{S}}) \\ &= -\mathop{E}_{f_i\in\mathcal{F}^u}[cos(f_i, \theta^{c_i}_{\mathcal{S}})/\tau] + \mathop{E}_{j\in[B]}\left[\log \sum_{k\neq j, k\in[B]} \exp(cos(\theta^j_{\mathcal{S}}, \theta^k_{\mathcal{S}})/\tau)\right] \end{aligned} \tag{4}$$

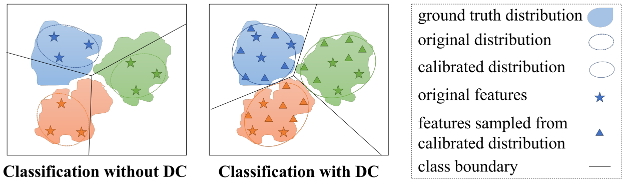

Figure 3. The diagram showcases two contrasting scenarios in the finetuning of a pretrained model. On the left, finetuning with a limited number of sample features leads to model overfitting. On the right, employing features sampled from a calibration distribution for finetuning the pretrained model demonstrates improved generalization.

where the balance weight λ is empirically set to 1, while the temperature scale τ is set to 0.07.

Finally, the loss function in Eq. (4) is directly optimized using gradient descent. After completing the optimization, the feature $\{f_{s_j}\}_{j\in[B]}$ with the highest similarity to $\theta^j_{\mathcal{S}}$ is identified. The corresponding data samples $\{\mathrm{x}_{s_j}\}_{j\in[B]}$ are selected as the subset $\mathcal{P}^u_{\mathcal{S}}$ with selection strategy $\mathcal{S} = \{s_j\}_{j\in[B]}$.

3.3. Distribution Calibration

Due to the typically limited number of labeled samples selected within the pretraining-finetuning paradigm, a substantial distributional discrepancy arises between the chosen subset and the entire data pool. This scenario, if unaddressed, may result in the overfitting of model parameters during the finetuning process, ultimately diminishing overall performance. In this section, we will not only use the labeled samples from Sec. 3.2 but also leverage information about the number of classification categories and other important information contained in a large number of unlabeled pretrained features. To mitigate distributional bias, our method involves the selection of pseudo-labeled data that exhibit reliability and approximate the overarching distribution accurately, as shown in Fig. 3.

Tukey's Ladder of Powers Transformation. To make the feature distribution more Gaussian-like, we first transform the features in $\mathcal{F}^u$ using Tukey's Ladder of Powers transformation [14]. Tukey's Ladder of Powers transformation belongs to a family of power transformations known for reducing distribution skewness and making distributions more Gaussian-like. This step is a prerequisite for the subsequent generation of features aligned with calibrated statistics conforming to a Gaussian distribution. Tukey's Ladder of Powers transformation is formulated as:

$$\hat{x} = \begin{cases} x^{\lambda}, & \text{if } \lambda \neq 0 \\ \log x, & \text{if } \lambda = 0 \end{cases} \tag{5}$$

where λ is a hyper-parameter used to control the distribution correction. The original feature can be recovered by setting

λ as 1. Decreasing λ makes the distribution less positively skewed and vice versa.

Statistics of Pseudo-Category. Our initial procedure entails clustering the normalized high dimensional features, referred to as $\mathcal{F}^u$, based on the true number of categories. Subsequently, pseudo-labels are assigned to each feature, drawing upon the pool of labeled features $\mathcal{F}^l_{\mathcal{S}}$ for reference, where $\mathcal{F}^l_{\mathcal{S}} = \{f_{s_j}, y_{s_j}\}_{j\in[B]}$. Thus we get the pool of pseudo-labeled features $\mathcal{F}^u_{\mathcal{C}} = \{f_{\mathcal{C}_j}, y_{\mathcal{C}_j}\}_{j\in[N]}$, where $\mathcal{C}$ is the cluster method. Assuming there are a total of K categories in $\mathcal{F}^u_{\mathcal{C}}$, the feature pool for category $y_{\mathcal{C}_j}$ is represented as $\mathcal{F}^i_{\mathcal{C}} = \{f_{\mathcal{C}_j}\}_{y_{\mathcal{C}_j}=i, j\in[N], i\in[K]}$. Most of the features corresponding to the same real label in $\mathcal{F}^l_{\mathcal{S}}$ reside in the same pseudo-category, correcting i to the corresponding real label.

$$
\begin{aligned}
\mathrm{E}_i &= \frac{\sum_{k=1}^{|\mathcal{F}^i_{\mathcal{C}}|} f_k}{|\mathcal{F}^i_{\mathcal{C}}|}, \quad s.t. \ \ f_k \in \mathcal{F}^i_{\mathcal{C}} \\
\mathrm{S}^2_i &= \frac{1}{|\mathcal{F}^i_{\mathcal{C}}| - 1} \sum_{k=1}^{|\mathcal{F}^i_{\mathcal{C}}|} (f_k - \mathrm{E}_i)(f_k - \mathrm{E}_i)^T
\end{aligned}
\tag{6}
$$

According to Eq. (6), we can get the mean and covariance of each pseudo-category feature pool. A similar process can be employed to calculate the mean of features corresponding to category i within $\mathcal{F}^l_{\mathcal{S}}$, denoted as μ^l_i.

Statistics Calibration. To eliminate the effect of outliers on the feature mean E_i, we recalculate the mean of the features by taking 90% of the features that are closest to the mean, denoted as $\hat{\mu_i}$. The calibrated mean and covariance are denoted as μ_i and σ^2_i, respectively, formulated as:

$$
\mu_i = (1-\beta)\hat{\mu_i} + \beta\mu^l_i \ \ , \ \ \sigma^2_i = \mathrm{S}^2_i + \xi \tag{7}
$$

where ξ is a hyper-parameter that determines the degree of dispersion of features sampled from the calibrated distribution. The term $(1-\beta)$ represents the degree to which the pseudo-category center contributes to the central information of the sample.

$$
\beta = 1 - \mathrm{e}^{-\alpha r}, \quad s.t. \ \ r \in [0, 100] \tag{8}
$$

In Eq. (8), α is a hyper-parameter whose value determines the parameter β in Eq. (7). The parameter β represents the degree to which the true labeled data acquired through data selection contributes to the central information of the sample. The sampling ratio is denoted as $r\%$. For instance, when r equals 1, the sampling ratio is 1%. Based on our understanding of the data distribution and experimental findings, we set the hyperparameter α to 0.7, 0.07, and 0.14 for CIFAR10, CIFAR100, and ImageNet, respectively.

Feature Generation and Filtering. With a set of calibrated statistics $\mathbb{S}_y = \{\mu_y, \sigma^2_y\}$ for class $y = i$ in a target task, we generate a set of feature vectors with label y by sampling from the calibrated Gaussian distributions defined as:

$$
\mathbb{G}_y = \{(f^g_y, y) \mid f^g_y \sim \mathcal{N}(\mu_y, \sigma^2_y), \forall(\mu_y, \sigma^2_y) \in \mathbb{S}_y\} \tag{9}
$$

The process of feature generation relies on creating n-fold features within the same class, and this is dependent on the number of features and classes in the labeled feature pool $\mathcal{F}^l_{\mathcal{S}}$. We identify the real features in the unlabeled pool that have the highest similarity to the generated features, formulated as:

$$
\mathbb{R}_y = \{(\hat{f}^g_y, y) \mid \max\ cos(f^g_y, \hat{f}^g_y), \forall(f^g_y, y) \in \mathbb{G}_y, \exists \hat{f}^g_y \in \mathcal{F}^u\} \tag{10}
$$

The data samples $\hat{\mathbb{R}}_y$ corresponding to $\mathbb{R}_y$, in conjunction with the pool of labeled data samples $\mathcal{P}^l_{\mathcal{S}}$, constitute the extended labeled pool $\mathcal{P}_{EL}$. This pool is subsequently utilized as the finetuning dataset for the pretrained model.

Algorithm 1: Pseudo-code for ActiveDC

```
input  : the unlabeled feature pool $\mathcal{F}^u$
output: the extended labeled pool $\mathcal{P}_{EL}$
// Data Selection by optimizing Eq. (4)
$\mathcal{F}^u_{\mathcal{S}}, \mathcal{P}^u_{\mathcal{S}} \leftarrow$ ActiveFT ($\mathcal{F}^u$);
// Oracle label data selected
$\mathcal{F}^l_{\mathcal{S}}, \mathcal{P}^l_{\mathcal{S}} \leftarrow$ oracleLabel ($\mathcal{F}^u_{\mathcal{S}}$, $\mathcal{P}^u_{\mathcal{S}}$);
// Powers Transformation As Per Eq. (5)
$\mathcal{F}^u \leftarrow$ Transform ($\mathcal{F}^u$);
// Clustering via K-Means
$\{\mathcal{F}^i_{\mathcal{C}}\}_{i\in[K]} \leftarrow$ Cluster ($\mathcal{F}^u$);
// Calibrate Statistics by Eqs. (6) and (7)
$\mu_i, \sigma^2_i \leftarrow$ Calibrate ($\{\mathcal{F}^i_{\mathcal{C}}\}_{i\in[K]}$, $\mathcal{F}^l_{\mathcal{S}}$);
// Initialize pseudo-label feature pool
$\mathbb{R}_{y,\forall y\in[K]} \leftarrow \{\}$;
for $i \leftarrow 0$ to $K$ do
    // Feature Generation As Per Eq. (9)
    $\{f_{gene.}, y = i\}_{\times n} \sim \mathcal{N}(\mu_{i,} \sigma^2_i)$;
    // Real feat. most similar per Eq. (10)
    $\{f_{real}, y = i\}_{\times n} \leftarrow$ Sim ($\{f_{gene.}, y = i\}_{\times n}$);
    // Filter harmful features
    $\mathbb{R}_y \leftarrow \mathbb{R}_y \bigcup$ Filter ($\{f_{real}, y = i\}_{\times n}$);
end
// Constitute the extended labeled pool
$\mathcal{P}_{EL} \leftarrow \mathcal{P}^l_{\mathcal{S}} \bigcup \hat{\mathbb{R}}_{y,\forall y\in[K]}$;
```

Additionally, it is essential to highlight that the generated features undergo a filtering process. The feature pool consisting of all the features in $\mathbb{R}_y$ is denoted as $\mathcal{F}^u_{\mathcal{G}} = \{\hat{f}^g_y\}_{(\hat{f}^g_y, y)\in\mathbb{R}_y}$. The extended feature pool $\mathcal{F}^u_E$ ($\mathcal{F}^u_{\mathcal{S}} \bigcup \mathcal{F}^u_{\mathcal{G}}$) is also associated with the extended data subset in $\mathcal{P}_{EL}$, with the corresponding distribution over $\mathcal{F}^u_E$ in the feature space denoted as p_{f_E}. The Earth Mover's Distance (EMD) metric [29] is employed as a quantitative measure for assessing the dissimilarity between a subset distribution p_{f_E} and the overall distribution p_{f_u}. Its application enables the identification and elimination of generative features that pose detriment to the overall distribution. More details about filtering process are available in the supplementary materials. Our method is summarised in Algorithm 1.

4. Experiments

Our method is evaluated on three image classification datasets of different scales with different sampling ratios. We compare the evaluation results with several baseline algorithms, traditional active learning algorithms, and ActiveFT, an efficient active finetuning algorithm. These will be presented in Sec. 4.1 and Sec. 4.2. We provide both qualitative and quantitative analysis of our method in Sec. 4.3. Finally, we investigate the role of different modules and different values of hyperparameters in our method in Sec. 4.4. The experiments were conducted using two GeForce RTX 3090 (24GB) GPUs, employing the DistributedDataParallel technique to accelerate the finetuning process.

4.1. Experiment Settings

Dataset and Metric. Our method, ActiveDC, is applied and evaluated on three well-established image classification datasets of varying classification scales. These datasets, namely CIFAR10, CIFAR100 [24], and ImageNet-1k [30], each present distinct characteristics. CIFAR10 and CIFAR100 consist of 60,000 images, with 10 and 100 categories, respectively. Both datasets use 50,000 images for training and 10,000 for testing. In contrast, ImageNet-1k has 1,000 categories with 1,281,167 training images and 50,000 validation images. The performance evaluation of our method is conducted using the *Top-1 Accuracy* metric.

Baselines. We compare our method to five traditional active learning methods and four baseline methods, including the strong baseline ActiveFT, the first efficient method applied to active finetuning tasks. The five active learning algorithms, namely CoreSet [31], VAAL [34], Learn-Loss [44], TA-VAAL [21], and ALFA-Mix [27], have been extended and adapted to the active finetuning task. These selected active learning methods cover both diversity-based and uncertainty-based strategies within the active learning domain. The set of baseline methods comprises four distinct techniques, including random selection, K-Center-Greedy, K-Means, and ActiveFT algorithm [41]. These baseline methods serve as crucial reference points for the comprehensive evaluation of active finetuning approaches.

- **Random:** A straightforward baseline method involves the random selection of B samples from the unlabeled pool, where B is denoted as the annotation budget.
- **FDS:** $a.k.a$ K-Center-Greedy algorithm. This method entails the selection of the next sample feature that is the farthest from the current selections. It is designed to minimize the disparity between the expected loss of the entire pool and that of the selected subset.
- **K-Means:** In this context, the value of K is set to the budget size, denoted as B. Overclustering is a strategy in which more clusters or subgroups are intentionally created than the expected number of distinct classes or groups in the data. Many studies in the unsupervised domain [6, 19, 35] have shown that overclustering leads to better performance.
- **ActiveFT:** A method that constructs a representative subset from an unlabeled pool, aligning its distribution with the overall dataset while optimizing diversity through parametric model optimization in a continuous space.

Implementation details. In unsupervised pretraining phase, we adopt DeiT-Small architecture [36] pretrained within the DINO framework [7], a well-established and effective choice on the ImageNet-1k dataset [30]. For consistency throughout the process, all images are resized to 224×224. In the data selection phase, the parameters denoted as θ_S are optimized employing the Adam optimizer [23] with a learning rate of $1\mathrm{e}{-3}$ until convergence. In the distribution calibration phase, the unsupervised clustering method and similarity retrieval mechanisms employed primarily rely on the FAISS (Facebook AI Similarity Search) library. Specifically, we employ the GPU-accelerated variant of K-Means for clustering and cosine similarity as the similarity metric. We use a full traversal search approach in FAISS for efficient retrieval. In the supervised finetuning phase, the DeiT-Small model follows the established protocol outlined in reference [7]. The implementation of supervised finetuning is based on the official codebase of DeiT. Further elaboration on the experiments is provided in the *supplementary materials*.

4.2. Overall Results

Our reported results are derived from meticulous averaging across three independent experimental runs, presented in Tab. 1. The traditional active learning methods tend to falter within the pretraining-finetuning paradigm, as documented in prior works [4, 16, 41]. In stark contrast, our method, ActiveDC, exhibits superior performance across all three datasets, even at varying sampling ratios. Particularly noteworthy is the substantial performance improvement observed with a lower sampling ratio. This improvement can be attributed to our method's capability to not only select the most representative samples but also effectively calibrate the distribution of the sampled data. This observed phenomenon carries practical significance, especially in scenarios where the number of samples utilized in the pretraining-finetuning paradigm is considerably smaller than the size of the available pool. Such efficiency gains contribute to substantial cost savings in terms of annotation expenditures. For instance, our method demonstrates a sig-

Table 1. The experiments were conducted on different scale datasets with different sampling ratios. We report the average of multiple trials. The symbol “-” is used to indicate not applicable (N/A). Specifically, this symbol indicates cases where active learning cannot be applied because the sample size is too small.

Methods	CIFAR10					CIFAR100				ImageNet			
	0.1%	0.2%	0.5%	1%	2%	1%	2%	5%	10%	0.5%	1%	2%	5%
Random	36.7	49.3	77.3	82.2	88.9	14.9	24.3	50.8	69.3	29.9	45.1	53.0	64.3
FDS	27.6	31.2	64.5	73.2	81.4	8.1	12.8	16.9	52.3	19.9	26.7	42.3	55.5
K-Means	40.3	58.8	83.0	85.9	89.6	17.6	31.9	42.4	70.7	37.1	50.7	55.7	62.2
CoreSet [31]	-	-	-	81.6	88.4	-	30.6	48.3	62.9	-	-	-	61.7
VAAL [34]	-	-	-	80.9	88.8	-	24.6	46.4	70.1	-	-	-	64.0
LearnLoss [44]	-	-	-	81.6	86.7	-	19.2	38.2	65.7	-	-	-	63.2
TA-VAAL [21]	-	-	-	82.6	88.7	-	34.7	46.4	66.8	-	-	-	64.3
ALFA-Mix [27]	-	-	-	83.4	89.6	-	35.3	50.4	69.9	-	-	-	64.5
ActiveFT [41]	47.1	64.5	85.0	88.2	90.1	26.1	40.7	54.6	71.0	36.8	50.1	54.2	65.3
ActiveDC (ours)	**61.3**	**73.1**	**87.3**	**88.9**	**90.3**	**34.5**	**54.6**	**71.9**	**74.3**	**50.9**	**56.3**	**60.1**	**68.2**

Table 2. Generality of Distributed Calibration Module.

Methods	CIFAR10 (0.5%)		CIFAR100 (5%)		ImageNet (1%)	
	w.o.DC	w.DC	w.o.DC	w.DC	w.o.DC	w.DC
Random	77.3	**86.2**	50.8	**70.1**	45.1	**54.6**
FDS	64.5	**78.3**	16.9	**33.4**	26.7	**40.2**
K-Means	83.0	**86.8**	42.4	**66.8**	50.7	**56.8**
ActiveFT	85.0	**87.3**	54.6	**71.9**	50.1	**56.3**

Table 3. Data selection efficiency of different methods.

Methods	2%	5%	10%
CoreSet	1h57m	7h44m	20h38m
VAAL	7h52m	12h13m	36h24m
LearLoss	20m	1h37m	9h09m
K-Means	16.6s	37.0s	70.2s
ActiveFT	12.6s	21.9s	37.3s
ActiveDC (ours)	2m40s	4m30s	7m20s

nificant accuracy increase of more than 10% when applied to the CIFAR10 dataset with a sampling rate of less than 0.2%. Similarly, when applied to the CIFAR100 dataset with a sampling rate of less than 5%, our method exhibits a similar improvement. This performance is in comparison to the strong benchmark ActiveFT [41].

4.3. Analysis

Generality of our Method. Our Distributed Calibration Module has the inherent flexibility to seamlessly integrate with a variety of active finetuning selection strategies. As indicated in Tab. 2, we have conducted experiments on datasets of varying scales. The findings demonstrate the efficacy of our method across diverse diversity-based selection strategies.

Data Selection Efficiency. Efficient data selection is crucial, requiring both time-efficient and effective methods. In Tab. 3, we compare the time needed to select different ratios of training samples from CIFAR100. Traditional active learning involves repeated model training and data sampling, with training being the major time-consuming factor. In contrast, active finetuning methods select all samples in a single pass, eliminating the need for iterative model retraining in the selection process. Our method, ActiveDC, increases processing time slightly compared to ActiveFT due to clustering and subsequent similarity retrieval after feature generation. However, the additional time, measured in minutes, is negligible compared to the significant performance improvements.

Visualization of Selected Samples. Figure 4 depicts the visualization of features extracted from the CIFAR10 training dataset. We utilize t-distributed Stochastic Neighbor Embedding (t-SNE) [39] for dimensionality reduction, employing distinct colors to differentiate feature categories. The pentagram symbolizes a 0.1% sample selected based on its distributional similarity to the overall sample, while the triangle represents a pseudo-labeled sample chosen after the distribution calibration. This visualization offers an effective way to intuitively understand the impact of our method on sample selection in terms of distribution calibration. For a comprehensive understanding, we recommend consulting Fig. 3 in conjunction with this visualization.

4.4. Ablation Study

Choices of Power for Tukey’s Transformation. In our analysis, we systematically investigate the impact of varying the hyperparameter λ in Eq. (5) on classification accuracy during Tukey’s Ladder of Powers transformation process. Notably, the most favorable accuracy results were consistently achieved when setting λ to 0.5 across all three datasets. Figure 5 illustrates the accuracy outcomes when finetuning the model with a 1% data sample extracted from the CIFAR100 dataset.

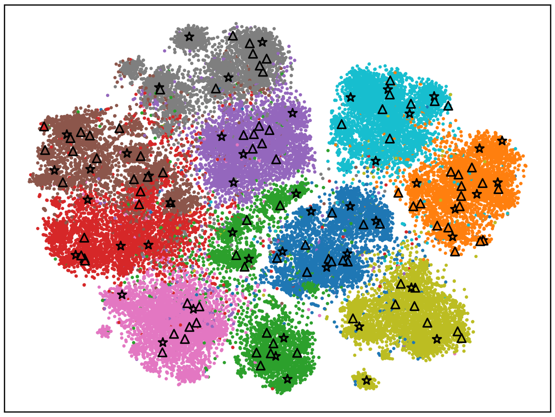

Figure 4. t-SNE Embeddings of CIFAR10: We visualize the embedding of selected samples labeled by the oracle (represented by a pentagram) and distribution calibration samples via ActiveDC (represented by a triangular shape) at a sampling ratio of 0.1%. Best viewed in color.

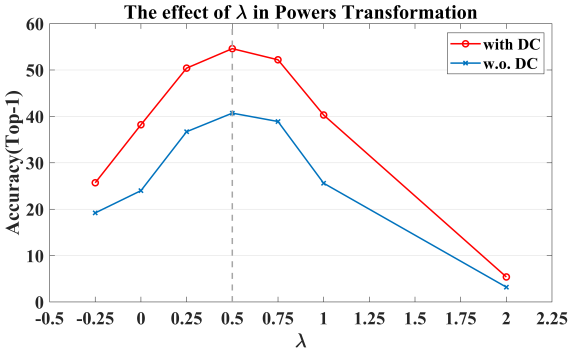

Figure 5. The effect of λ: The top fold (in red) represents finetuning accuracy with statistical calibration, while the lower fold (in blue) represents finetuning accuracy without statistical calibration. These results are obtained using different values of hyperparameter λ in Eq. (5).

Table 4. The effect of different values of ξ.

hyperparameter ξ	**-0.1**	**0**	**0.1**	**0.2**	**0.3**
CIFAR10 (0.5%)	85.5	86.4	86.9	**87.3**	70.2
CIFAR100 (5%)	65.3	68.6	71.5	**71.9**	61.8
ImageNet (1%)	48.7	50.6	56.0	**56.3**	50.2

Hyperparameter Tuning for Statistics Calibration. Based on our understanding of the data distribution and experimental findings, we set the hyperparameter α in Eq. (8) to 0.7, 0.07, and 0.14 for CIFAR10, CIFAR100, and ImageNet, respectively. For details regarding the selection of α values in our experiments, please refer to the supplementary material. We analyze the effect of different values of the hyperparameter ξ on the classification accuracy during the statistical calibration process, as shown in Tab. 4. The hyperparameter ξ in Eq. (7) determines the degree of dispersion of features sampled from the calibrated distribution.

Table 5. The effect of the number of generated features.

number of gene.	1×	2×	3×	4×
CIFAR10 (0.5%)	85.9	**87.3**	85.1	81.0
CIFAR100 (5%)	64.5	**71.9**	65.3	57.1
ImageNet (1%)	54.7	**56.3**	56.1	55.0

Table 6. The effect of different modules in Distributed Calibration.

Distribution Calibration			CIFAR10	
Powers Transformation	Statistics Calibration	Feature Filtering	0.1%	0.2%
×	×	×	47.1	64.5
✓	×	×	50.3	66.8
✓	✓	×	59.5	70.6
✓	✓	✓	**61.3**	**73.1**

Number of Generated Features. The number of generated features is twice the count of labeled features, as shown in Tab. 5. For each labeled feature, we generate two corresponding features, each adhering to the same calibrated distribution as the original feature. It is important to note that an excessive proliferation of pseudo-labeled data can lead to a decrease in accuracy. This phenomenon can be attributed to the presence of mislabeled data that is inadequately filtered out during the process described in Sec. 3.3.

The effect of different module in DC. To investigate the influence of various sub-modules in distribution calibration on performance, we systematically conducted incremental experiments with these sub-modules, as detailed in Tab. 6. The rationale behind adopting an incremental manner lies in the interdependence of subsequent modules on their antecedent counterparts for functionality. The experimental findings indicate that each submodule contributes positively to a certain extent.

5. Conclusion

This work proposes a novel method named ActiveDC, designed for data selection and distribution calibration of active finetuning tasks. The method consists of two crucial steps: Data Selection and Distribution Calibration. In the Distribution Calibration step, we leverage a substantial volume of unlabeled pretrained features to extract insights into class distribution and robustly calibrate the statistical information by ingeniously combining it with information from the labeled samples. Subsequently, we select pseudo-labeled data points that demonstrate reliability and provide an accurate approximation of the overall data distribution. Extensive experiments have demonstrated its effectiveness and significance. For future work, we will concentrate on optimizing efficiency, making it applicable to a broader range of application scenarios.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (No. 62302031, No. 62176017), the Zhejiang Provincial Natural Science Foundation of China (No. LQ23F020024), and the Key R&D Program of Zhejiang Province (No. 2024C01020).

References

[1] Sharat Agarwal, Himanshu Arora, Saket Anand, and Chetan Arora. Contextual diversity for active learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pages 137–153. Springer, 2020. 3

[2] Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations*, 2020. 1, 3

[3] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEit: BERT pre-training of image transformers. In *International Conference on Learning Representations*, 2022. 3

[4] Javad Zolfaghari Bengar, Joost van de Weijer, Bartlomiej Twardowski, and Bogdan Raducanu. Reducing label effort: Self-supervised meets active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1631–1639, 2021. 2, 3, 6

[5] Vivien Cabannes, Leon Bottou, Yann Lecun, and Randall Balestriero. Active self-supervised learning: A few low-cost relationships are all you need. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16274–16283, 2023. 1

[6] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020. 3, 6

[7] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 3, 4, 6

[8] Liangyu Chen, Yutong Bai, Siyu Huang, Yongyi Lu, Bihan Wen, Alan Yuille, and Zongwei Zhou. Making your first choice: To address cold start problem in medical active learning. In *Medical Imaging with Deep Learning*, 2023. 3

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 1, 3

[10] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15750–15758, 2021. 3

[11] X Chen, S Xie, and K He. An empirical study of training self-supervised vision transformers. In *CVF International Conference on Computer Vision (ICCV)*, pages 9620–9629, 2021. 3

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 3

[13] Sebastian Farquhar, Yarin Gal, and Tom Rainforth. On statistical bias in active learning: How and when to fix it. In *International Conference on Learning Representations*, 2021. 1, 2

[14] Vaibhav Ganatra. Logarithm-transform aided gaussian sampling for few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 247–252, 2023. 4

[15] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020. 3

[16] Guy Hacohen, Avihu Dekel, and Daphna Weinshall. Active learning on a budget: Opposite strategies suit high and low budgets. In *International Conference on Machine Learning*, pages 8175–8195. PMLR, 2022. 2, 3, 6

[17] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 1, 3

[18] 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. 3

[19] Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9865–9874, 2019. 6

[20] Ajay J Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class active learning for image classification. In *2009 ieee conference on computer vision and pattern recognition*, pages 2372–2379. IEEE, 2009. 3

[21] Kwanyoung Kim, Dongwon Park, Kwang In Kim, and Se Young Chun. Task-aware variational adversarial active learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8166–8175, 2021. 3, 6, 7

[22] SangMook Kim, Sangmin Bae, Hwanjun Song, and Se-Young Yun. Re-thinking federated active learning based on inter-class diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3944–3953, 2023. 1

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[24] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6

[25] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9): 1–35, 2023. 1

[26] Mengyao Lyu, Jundong Zhou, Hui Chen, Yijie Huang, Dongdong Yu, Yaqian Li, Yandong Guo, Yuchen Guo, Liuyu Xiang, and Guiguang Ding. Box-level active detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23766–23775, 2023. 1

[27] Amin Parvaneh, Ehsan Abbasnejad, Damien Teney, Gholamreza Reza Haffari, Anton Van Den Hengel, and Javen Qinfeng Shi. Active learning by feature mixing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12237–12246, 2022. 3, 6, 7

[28] Aayush J Rana and Yogesh S Rawat. Hybrid active learning via deep clustering for video action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18867–18877, 2023. 1

[29] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. A metric for distributions with applications to image databases. In *Sixth international conference on computer vision (IEEE Cat. No. 98CH36271)*, pages 59–66. IEEE, 1998. 6

[30] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 6

[31] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018. 1, 3, 6, 7

[32] Changjian Shui, Fan Zhou, Christian Gagné, and Boyu Wang. Deep active learning: Unified and principled method for query and training. In *International Conference on Artificial Intelligence and Statistics*, pages 1308–1318. PMLR, 2020. 3

[33] Mannat Singh, Quentin Duval, Kalyan Vasudev Alwala, Haoqi Fan, Vaibhav Aggarwal, Aaron Adcock, Armand Joulin, Piotr Dollár, Christoph Feichtenhofer, Ross Girshick, et al. The effectiveness of mae pre-pretraining for billion-scale pretraining. *arXiv preprint arXiv:2303.13496*, 2023. 3

[34] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5972–5981, 2019. 3, 6, 7

[35] Thomas Stegmüller, Tim Lebailly, Behzad Bozorgtabar, Tinne Tuytelaars, and Jean-Philippe Thiran. Croc: Cross-view online clustering for dense visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7000–7009, 2023. 6

[36] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021. 6

[37] Toan Tran, Thanh-Toan Do, Ian Reid, and Gustavo Carneiro. Bayesian generative active deep learning. In *International Conference on Machine Learning*, pages 6295–6304. PMLR, 2019. 1

[38] Samyakh Tukra, Frederick Hoffman, and Ken Chatfield. Improving visual representation learning through perceptual understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14486–14495, 2023. 3

[39] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (11), 2008. 7

[40] Yichen Xie, Mingyu Ding, Masayoshi Tomizuka, and Wei Zhan. Towards free data selection with general-purpose models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 3

[41] Yichen Xie, Han Lu, Junchi Yan, Xiaokang Yang, Masayoshi Tomizuka, and Wei Zhan. Active finetuning: Exploiting annotation budget in the pretraining-finetuning paradigm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23715–23724, 2023. 2, 3, 4, 6, 7

[42] Lihe Yang, Zhen Zhao, Lei Qi, Yu Qiao, Yinghuan Shi, and Hengshuang Zhao. Shrinking class space for enhanced certainty in semi-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16187–16196, 2023. 1

[43] Shuo Yang, Lu Liu, and Min Xu. Free lunch for few-shot learning: Distribution calibration. In *International Conference on Learning Representations*, 2021. 1, 2

[44] Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 93–102, 2019. 3, 6, 7

[45] Aojun Zhou, Yang Li, Zipeng Qin, Jianbo Liu, Junting Pan, Renrui Zhang, Rui Zhao, Peng Gao, and Hongsheng Li. Sparsemae: Sparse training meets masked autoencoders. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16176–16186, 2023. 3

[46] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *International Conference on Learning Representations (ICLR)*, 2022. 1, 3