Rethinking Uncertainty in Active Learning for Image Classification: When Confidence Is Misleading

Zilong Ling
e125111005@stu.ahu.edu.cn
Anhui Provincial International Joint Research Center for Advanced Technology in Medical Imaging, School of Computer Science and Technology, Anhui University
Hefei, China

Huabin Wang
wanghuabin@ahu.edu.cn
Anhui Provincial International Joint Research Center for Advanced Technology in Medical Imaging, School of Computer Science and Technology, Anhui University
Hefei, China

Liang Du
duliang@sxu.edu.cn
School of Computer and Information Technology, Shanxi University
Taiyuan, China

Xuejun Li
xjli@ahu.edu.cn
Anhui Provincial International Joint Research Center for Advanced Technology in Medical Imaging, School of Computer Science and Technology, Anhui University
Hefei, China

Peng Zhou*
zhoupeng@ahu.edu.cn
Anhui Provincial International Joint Research Center for Advanced Technology in Medical Imaging, School of Computer Science and Technology, Anhui University
Hefei, China

Abstract

Active image classification is a fundamental task in image processing. One of the most popular active image classification methods is the uncertainty-based method, which selects the uncertain images for querying annotations. Conventional uncertainty-based active learning methods often regard the class boundary samples as the uncertain samples. However, in this paper, we observe a new kind of uncertain samples in active learning, which the model often misclassifies with high confidence, and thus we call them *misleading samples*. Since the model often has high confidence in these samples, without the annotations, the model does not even know they are uncertain or difficult samples, not to mention that the model knows itself makes mistakes on them. Therefore, misleading samples are much more difficult to identify, and conventional active learning methods often ignore them. To tackle this problem, we theoretically analyze how they were generated, and inspired by multi-modal learning, we discover the inconsistency property of them in the original spatial modal and frequency modal. Based on this analysis, we design a simple yet effective bi-modal method to identify the misleading samples. Then, we integrate it into an uncertainty-based framework to discover both kinds of uncertain samples. The extensive experiments show that our method discovers uncertain samples more comprehensively and performs better in the image classification task. Code is available at: https://github.com/Luming580/BUAL.

*Corresponding author.

MM '26, Rio de Janeiro, Brazil

ACM ISBN 979-8-4007-2213-4/2026/11
https://doi.org/10.1145/3767308.3836216

CCS Concepts

• **Computing methodologies** → **Active learning settings**.

Keywords

Active Learning, Image Classification, Bi-modal Learning

ACM Reference Format:
Zilong Ling, Huabin Wang, Liang Du, Xuejun Li, and Peng Zhou. 2026. Rethinking Uncertainty in Active Learning for Image Classification: When Confidence Is Misleading. In *Proceedings of the 34th ACM International Conference on Multimedia (MM '26), November 10–14, 2026, Rio de Janeiro, Brazil*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3767308.3836216

1 Introduction

Deep learning has achieved remarkable success in computer vision tasks, largely attributable to the availability of large-scale and high-quality annotated data. However, in many real-world applications, such as medical image analysis and autonomous driving perception, obtaining accurate annotations is not only costly but also extremely time-consuming. To tackle the problem of the acquisition of large-scale labeled data, active learning is widely studied [7, 22, 42]. Active learning selectively queries a small number of informative samples for annotation and trains the model, enabling performance comparable to fully supervised methods. The key point of active learning is to design effective query strategies that identify the most informative samples from a large number of unlabeled data to maximize model performance gains while minimizing the cost of labeling.

Conventional active learning strategies can be roughly categorized into four classes: uncertainty-based methods [16, 28, 44, 48, 51], representativeness-based methods [14, 30, 34], hybrid methods [2, 9], and adversarial methods [41, 49]. Among them, the uncertainty-based method is one of the most popular methods

in active learning. It tries to select the most uncertain data for annotations, because the model can hardly handle the uncertain data by itself and thus needs human help. Conventional uncertainty-based active learning methods regard the samples near the decision boundary or the samples with low confidence as uncertain data [28, 36, 44]. However, in our study, after carefully rethinking the uncertain data, we observe that there are two kinds of uncertain data in image classification: 1) *The data that the model knows are uncertain.* These data often stay on the boundary of two classes, and thus the model often has low confidence in these data, no matter whether the model can classify them correctly. Conventional uncertainty-based active learning methods often focus on this kind of uncertain data. 2) *The data that the model does NOT know are uncertain.* The model often classifies them incorrectly, but with high confidence. Therefore, these data may stay deep in another class. Figure 1 shows a simple example of t-SNE on two classes in the Cifar10 dataset. The first kind of uncertain data is the boundary data, which are marked in the purple circle. The second kind of uncertain data are the orange samples but in the blue class, and the blue samples in the orange class, which are marked by green circles. Conventional uncertainty-based methods often ignore the second kind of uncertain data because this kind of data is much more difficult to discover than the first kind, as the model does not know it cannot handle them correctly.

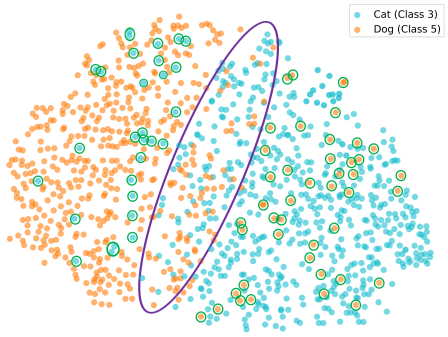
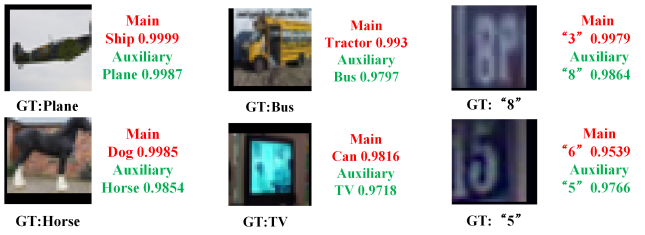


Figure 1: t-SNE visualization of the samples on Cifar10. The purple region denotes uncertain samples located near the decision boundary, while the green circles represent misleading samples, for which the model makes high-confidence but incorrect predictions.

In this paper, we focus on the second kind of uncertain data in image classification. Since the model often incorrectly classifies these data with high confidence and they may mislead the model training in turn, we call them *misleading samples*. Since the model does not know it cannot handle misleading samples correctly, it is difficult to identify them only by the model itself. To tackle this problem, we theoretically analyze how they were generated. Inspired by multi-modal learning, we discover the inconsistency property in the spatial and frequency modalities of the misleading samples. Specifically, we observe that an auxiliary classification model on the frequency modal of the weighted wavelet transform (WWT) [27, 31, 43] may be helpful in identifying misleading samples. From the perspective of perturbing a normal sample to turn it into a misleading sample, we provide a theorem to show that the main model in the original modal and its auxiliary model in the frequency modal often perform very differently on the misleading samples. Figure 2 shows some examples of misleading samples from the Cifar10, Cifar100, and SVHN datasets. The red fonts denote the prediction and its confidence of the main model, and the green fonts denote the prediction and confidence of the auxiliary model. We can see that the main model classifies them incorrectly with high confidence, but its auxiliary model often provides another classification result. Therefore, we can identify these misleading samples by checking the inconsistency between the two modalities. Since we identify misleading samples with the original spatial modal and the WWT frequency modal, we call our method Bi-modal Uncertainty-based Active Learning (BUAL).



(a) Cifar10. **(b) Cifar100.** **(c) SVHN.**

Figure 2: Misleading samples from different datasets. Red fonts indicate the predicted class and confidence produced by the spatial-domain main model, while green fonts denote the predicted class and confidence given by the frequency-domain auxiliary model.

Based on this observation and theoretical analysis, we design a simple yet effective bi-modal strategy to identify the misleading samples by defining their misleading scores. Besides, to identify the first type of uncertain data, which often lies at the boundaries of classes (we call them "*boundary samples*"), we apply a conventional uncertainty-based method to assign a boundary score. Then, we carefully design an adaptive weighted strategy to integrate the misleading score and boundary score, leading to a novel uncertainty-based active learning method. By combining the misleading score and the boundary score, we evaluate the uncertainty of the data more comprehensively.

The main contributions are summarized as follows:

- We introduce a new type of uncertain data in active image classification, termed misleadingness, to characterize samples that are predicted with high confidence yet incorrectly classified by the model. Conventional uncertainty-based active learning methods often ignore them.
- We propose a theoretical analysis of the generation of the misleading samples. Based on this theoretical analysis, we design a simple yet effective bi-modal method to select them. We also propose a novel fusion strategy to balance the misleading score and conventional boundary score to evaluate the uncertainty of data more comprehensively.
- Extensive experiments on multiple benchmark datasets demonstrate that our method often achieves state-of-the-art performance across different numbers of annotations, thereby validating the effectiveness of the proposed method.

2 Related Work

In this section, we briefly review some related work of active learning. One kind of the most widely-studied active learning is the batch mode active learning [6, 23, 35, 45, 50]. In batch mode active learning, given a dataset $\mathcal{X}$ with n samples, we select several important samples for querying the annotations in several iterative batches. More specifically, we divide the data into two groups: labeled samples $\mathcal{L}$ and unlabeled samples $\mathcal{U}$, where $\mathcal{L} \cup \mathcal{U} = \mathcal{X}$ and $\mathcal{L} \cap \mathcal{U} = \emptyset$. In each iterative batch, given a budget B, we need to select at most B samples (denoted as $\mathcal{S}$) from $\mathcal{U}$ and query the annotations of the selected B samples. Then, we move $\mathcal{S}$ from $\mathcal{U}$ to $\mathcal{L}$, i.e., $\mathcal{U} = \mathcal{U} - \mathcal{S}$ and $\mathcal{L} = \mathcal{L} \cup \mathcal{S}$. After that, we use the new labeled set $\mathcal{L}$ to train the model. Then, we go to the next iteration. This process will go on until the total budget runs out. When selecting $\mathcal{S}$ for annotation, active learning wishes to select important and informative samples, which are helpful to training the model. There are several selective strategies in active learning, which are roughly categorized into four classes: uncertainty-based methods [18, 28, 33, 44], representativeness-based methods [17, 30, 34, 39], hybrid methods [2, 9, 29, 40], and adversarial methods [20, 41, 49].

Since this paper focuses on the uncertainty of data, we briefly review some works of uncertainty-based methods. The basic idea of uncertainty-based approaches is to select uncertain samples that the classifier finds difficult to predict correctly. These methods can be further divided into Bayesian and non-Bayesian frameworks. In Bayesian active learning, probabilistic models such as Bayesian neural networks [8, 10, 11] and Gaussian processes [19] are employed to estimate predictive uncertainty. For example, Houlsby et al. [16] introduced the Bayesian Active Learning by Disagreement (BALD) algorithm, which selects instances based on information gain. Kirsch et al. [21] extended BALD to deep learning settings by proposing BatchBALD, which enables the joint selection and annotation of multiple instances. In contrast, non-Bayesian active learning methods often quantify uncertainty [18, 28, 32, 33, 44] using the posterior probabilities predicted by the target model. Yoo et al. [48] introduced a loss prediction module that estimates sample-wise prediction error as an uncertainty measure.

3 Method

Our method is an uncertainty-based method, which selects uncertain data for querying human annotations. Based on our observation, we categorize the data into three types: 1) *Normal samples*, which the model can classify correctly with high confidence. They often lie in or near the center of their class. 2) *Boundary samples*, which the model may predict correctly or incorrectly with low confidence. They often lie on the boundary of classes. 3) *Misleading samples*, which the model classifies incorrectly with high confidence. They often lie in or near the center of another class. It is obvious that the normal samples are easy samples, and the boundary samples and misleading samples are more difficult to handle. Therefore, we regard the boundary samples and misleading samples as *uncertain* samples. Conventional uncertainty-based methods only focus on the boundary samples while ignoring the misleading samples. Unfortunately, the misleading samples are much more difficult to identify than the boundary samples, because the model cannot distinguish them from the normal samples before obtaining their annotations. To this end, one main contribution of our method is to identify the misleading samples in active learning.

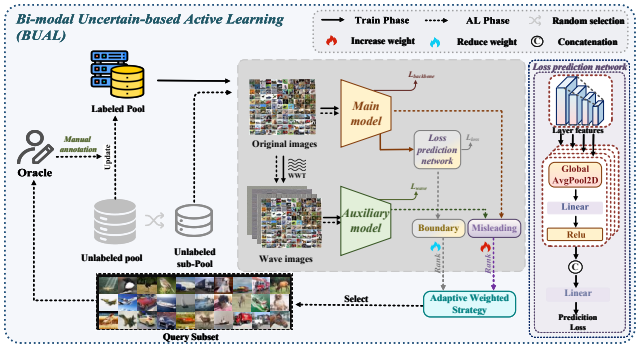


Figure 3: Overall architecture of BUAL. Given an image, the framework uses a main model for classification. Besides, the framework also applies a weighted wavelet transform (WWT) to obtain the frequency modal of an image and feed it into the auxiliary model. The framework identifies the misleading samples by comparing the difference between the output of the main model and the auxiliary model. Moreover, the framework applies a loss prediction network to discover the boundary samples. Then, an adaptive weighted strategy is designed to combine these two kinds of uncertain data and select the final samples for annotation.

Figure 3 illustrates the overall architecture of our method. Following the common setting of previous active learning methods [47, 48], we also use ResNet-18 [15] as a main model (denoted as $\mathcal{F}(\cdot)$) for classification. Notice that our method can also be applied to any other backbone classification networks easily. To discover the misleading samples, we design a same-structure auxiliary network (denoted as $\mathcal{A}(\cdot)$) but in the frequency domain, which will be introduced in Section 3.1. We also use a loss prediction network [48] to discover the boundary samples, which will be introduced in Section 3.2. Since uncertain data consists of both misleading samples and boundary samples, we carefully design an adaptive weighting strategy to combine the misleading samples and boundary samples, which will be introduced in Section 3.3. In the following, we will introduce each part in more detail.

3.1 Misleading Samples Selection

As discussed before, the model itself cannot identify the misleading samples because it cannot distinguish them from normal samples without annotations. To this end, we design an auxiliary model to discover the misleading samples. Since the main model predicts normal samples correctly but makes mistakes on the misleading samples, we wish the auxiliary model performs similarly on the normal samples but performs differently on the misleading samples. Notice that the normal samples are often the majority, which means the auxiliary model should perform similarly to the main model most time. Hence, we make the auxiliary model have the same structure as the main model. To make the auxiliary model predict differently from the main model on misleading samples, we perform a domain transformation on the images to obtain another modal. On one hand, since we wish the auxiliary model to have a consistent

prediction with the main model on the majority normal samples, the transformation should not be too complicated and is best as a linear transformation. On the other hand, existing deep learning models are prone to overfitting high-frequency noise or specific texture patterns [46], which can lead to high-confidence misclassifications under small perturbations of these features, causing misleading samples, and thus it should be a noise-robust transformation. To this end, we adopt a 2D weighted wavelet transform (denoted as $\mathcal{H}(\cdot)$) [27, 43] to transform the data into the frequency modal. In more detail, given an image $\mathbf{X} \in \mathbb{R}^{H\times W\times C}$ with width W, height H, and C channels, we flatten each channel into a vector $\mathbf{x}$ whose length is $H \times W$. The 2D weighted wavelet transform is equivalent to a linear transformation $\mathcal{H}(\mathbf{x}) = \mathbf{H}\mathbf{x}$, where $\mathbf{H}$ is a constant predefined transformation matrix, whose definition is introduced in Appendix A. Finally, we obtain this image in the frequency modal by reshaping the vector $\mathcal{H}(\mathbf{x})$ into a matrix and stacking all channels.

We train the main model and auxiliary model with the labeled set $\mathcal{L}$ separately. In more detail, consider an image $\mathbf{x}_i \in \mathcal{L}$ whose one-hot human annotation is $\mathbf{y}_i \in \{0,1\}^K$, where K is the number of classes. We feed $\mathbf{x}_i$ into the main model to obtain its prediction $\mathcal{F}(\mathbf{x}_i)$ and train the main model with cross-entropy loss $\mathcal{L}_{\text{ce}}(\mathbf{y}_i, \mathcal{F}(\mathbf{x}_i))$. Then we apply WWT to obtain the frequency modal $\mathcal{H}(\mathbf{x}_i)$ and feed it into the auxiliary model to obtain the prediction $\mathcal{A}(\mathcal{H}(\mathbf{x}_i))$. We train auxiliary network also by cross-entropy loss $\mathcal{L}_{\text{ce}}(\mathbf{y}_i, \mathcal{A}(\mathcal{H}(\mathbf{x}_i)))$. In the following, we introduce how to discover misleading samples using the two modalities.

3.1.1 Theoretical Motivations. Before introducing the method, we first provide a theoretical analysis of how these samples are generated. Figure 4 shows an example. Consider a normal sample $\mathbf{x}$ in the k-th class. Due to the noise in the data or the limited representational capacity of the classification model, in the learned latent space, $\mathbf{x}$ is perturbed across the boundary and deep into the j-th class, becoming a misleading sample $\mathbf{x}^*$.

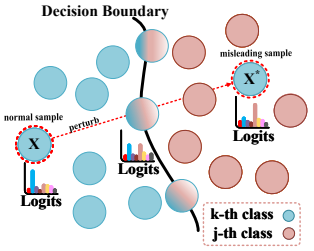


Figure 4: Due to noise perturbations, a sample x that originally belongs to the k-th class is pushed across the decision boundary and moves deep into the j-th class, becoming a misleading sample x*. The red arrow indicates the direction of the noise perturbation.

To identify such $\mathbf{x}^*$, we first analyze how to perturb the original $\mathbf{x}$ to $\mathbf{x}^*$. The largest difference between $\mathbf{x}$ and $\mathbf{x}^*$ is their predicted logits. Since the original $\mathbf{x}$ should belong to the k-th class, the k-th logit of $\mathbf{x}$ should be large. However, because $\mathbf{x}^*$ is a misleading sample, which means the model misclassifies it to the j-th class with high confidence, its j-th logit should be large, and thus the k-th logit should be small. Therefore, to efficiently move a sample from $\mathbf{x}$ to $\mathbf{x}^*$, the fastest way is to move it along the negative gradient of the k-th logit [43]. To simplify the theoretical analysis, we make some assumptions on data, and the experiments demonstrate that our method can work well on more complicated and practical cases. Given a dataset with K classes, we consider the case that each class follows a high-dimensional Gaussian distribution with identical covariance. More specifically, the samples in the k-th class follow the Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_k, \Sigma)$, where $\boldsymbol{\mu}_k$ is the mean vector of the k-th class and Σ is the covariance matrix. Then, Lemma 3.1 provides the perturbation direction. The proofs of all Lemmas and Theorems can be found in the Appendix.

Lemma 3.1. *Under the above assumption, given a sample $\mathbf{x}$ that belongs to the k-th class, the gradient direction of the k-th logit is $\Sigma^{-1}\boldsymbol{\mu}_k$.*

According to Lemma 3.1, we can perturb $\mathbf{x}$ along the negative gradient direction, i.e., $-\Sigma^{-1}\boldsymbol{\mu}_k$, as $\hat{\mathbf{x}} = \mathbf{x} - \eta\Sigma^{-1}\boldsymbol{\mu}_k$, where η is the step size of the perturbation. Since a misleading sample has a high confidence in another logit, it needs a large step size η.

In practice, the real misleading samples $\mathbf{x}^*$ may not be just the same as our ideal assumption $\hat{\mathbf{x}}$. We assume that the real $\mathbf{x}^*$ may be near our ideal one, which means $\|\mathbf{x}^* - \hat{\mathbf{x}}\|_2 \le \epsilon$ where ϵ is a small number. This is the generation process of a misleading sample $\mathbf{x}^*$ from a normal sample $\mathbf{x}$. If $\mathbf{x}$ is a normal sample, as discussed before, since we use a simple linear transformation $\mathcal{H}$ on it and $\mathcal{F}(\cdot)$ and $\mathcal{A}(\cdot)$ have the same structure, we assume that the prediction $\mathcal{F}(\mathbf{x})$ and $\mathcal{A}(\mathcal{H}(\mathbf{x}))$ should be similar, i.e., $\|\mathcal{F}(\mathbf{x}) - \mathcal{A}(\mathcal{H}(\mathbf{x}))\|_2$ should be small. Then, the following Theorem shows that even if $\|\mathcal{F}(\mathbf{x}) - \mathcal{A}(\mathcal{H}(\mathbf{x}))\|_2$ is small, when considering the perturbed misleading sample $\mathbf{x}^*$, $\|\mathcal{F}(\mathbf{x}^*) - \mathcal{A}(\mathcal{H}(\mathbf{x}^*))\|_2$ will be large with a high probability.

Theorem 3.2. *Given a dataset $\mathcal{X}$ under our Gaussian distribution assumption, $\mathcal{F}(\cdot)$ and $\mathcal{A}(\cdot)$ are trained on $\mathcal{X}$ and $\mathcal{H}(\mathcal{X})$ (with transformation matrix $\mathbf{H}$), respectively. Define δ as the largest difference between $\mathcal{F}(\mathbf{x})$ and $\mathcal{A}(\mathcal{H}(\mathbf{x}))$ for all normal samples, i.e., $\delta = \max_{\mathbf{x}} \|\mathcal{F}(\mathbf{x}) - \mathcal{A}(\mathcal{H}(\mathbf{x}))\|_2$. Define $\mathbf{c}_k = \left(\Sigma^{-1} - \mathbf{H}^\top(\mathbf{H}\Sigma\mathbf{H}^\top)^{-1}\mathbf{H}\right)\boldsymbol{\mu}_k$, $\tilde{\mu}^{(k)} = \mathbf{c}_k^\top\left(\frac{1}{2}\mathbf{I} - \eta\Sigma^{-1}\right)\boldsymbol{\mu}_k$, and $\tilde{\sigma}^{(k)} = \sqrt{\mathbf{c}_k^\top\Sigma\mathbf{c}_k}$. Considering a misleading sample $\mathbf{x}^*$ obtained by perturbing a normal sample $\mathbf{x}$, i.e., $\|\mathbf{x}^* - (\mathbf{x} - \eta\Sigma^{-1}\boldsymbol{\mu}_k)\| \le \epsilon$, we have:*

$$\mathbb{P}\{\|\mathcal{F}(\mathbf{x}^*) - \mathcal{A}(\mathcal{H}(\mathbf{x}^*))\|_2 \ge \delta\} \ge \min_{k=1,\dots,K}\left[\Phi\left(\frac{\tilde{\mu}^{(k)} - (L\epsilon+\delta)}{\tilde{\sigma}^{(k)}}\right) + \Phi\left(\frac{-\tilde{\mu}^{(k)} - (L\epsilon+\delta)}{\tilde{\sigma}^{(k)}}\right)\right], \tag{1}$$

where $\Phi(\cdot)$ denotes the cumulative distribution function (CDF) of the standard normal distribution $\mathcal{N}(0,1)$, and L is the largest singular value of $\mathbf{M}(\Sigma^{-1} - \mathbf{H}^\top(\mathbf{H}\Sigma\mathbf{H}^\top)^{-1}\mathbf{H})$, where $\mathbf{M} = [\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K]^T$.

Theorem 3.2 shows that $\|\mathcal{F}(\mathbf{x}^*) - \mathcal{A}(\mathcal{H}(\mathbf{x}^*))\|_2 \ge \delta \ge \|\mathcal{F}(\mathbf{x}) - \mathcal{A}(\mathcal{H}(\mathbf{x}))\|_2$ with a high probability. It means that, compared to normal samples, the main modal $\mathcal{F}(\cdot)$ and the auxiliary model $\mathcal{A}(\cdot)$ predict more differently on the misleading samples. To characterize how high the probability is, we need the following Lemma:

Lemma 3.3. *Denoting $\Phi(\cdot)$ as the CDF of $\mathcal{N}(0,1)$, given any positive constant a and b, $\Phi(\frac{x-a}{b}) + \Phi(\frac{-x-a}{b})$ will increase with increasing $|x|$.*

Now, consider $x = \tilde{\mu}^{(k)} = \mathbf{c}_k^\top\left(\frac{1}{2}\mathbf{I} - \eta\Sigma^{-1}\right)\boldsymbol{\mu}_k = \frac{1}{2}\mathbf{c}_k^\top\boldsymbol{\mu}_k - \eta\mathbf{c}_k^\top\Sigma^{-1}\boldsymbol{\mu}_k$. When η is large enough, $|\tilde{\mu}^{(k)}|$ will increase with increasing η. By

Lemma 3.3, we further have $\Phi\left(\frac{\tilde{\mu}^{(k)}-(L\epsilon+\delta)}{\tilde{\sigma}^{(k)}}\right)+\Phi\left(\frac{-\tilde{\mu}^{(k)}-(L\epsilon+\delta)}{\tilde{\sigma}^{(k)}}\right)$ will increase with increasing η. As discussed before, misleading samples often have a large perturbation η and thus the probability of $\|\mathcal{F}(\mathbf{x}^*)-\mathcal{A}(\mathcal{H}(\mathbf{x}^*))\|_2 \geq \|\mathcal{F}(\mathbf{x})-\mathcal{A}(\mathcal{H}(\mathbf{x}))\|_2$ is also large. Besides, it is easy to see that $\Phi\left(\frac{\tilde{\mu}^{(k)}-(L\epsilon+\delta)}{\tilde{\sigma}^{(k)}}\right)+\Phi\left(\frac{-\tilde{\mu}^{(k)}-(L\epsilon+\delta)}{\tilde{\sigma}^{(k)}}\right)$ decreases monotonically with ϵ, which means if the real misleading sample $\mathbf{x}^*$ is close to our ideal assumption $\hat{\mathbf{x}}$, this probability will also increase.

3.1.2 Method of Identifying Misleading Samples. Based on Theorem 3.2, we design a simple yet effective bi-modal method to identify the misleading samples. Theorem 3.2 tells us that, given a misleading sample, there is a high probability that $\|\mathcal{F}(\mathbf{x}^*)-\mathcal{A}(\mathcal{H}(\mathbf{x}^*))\|_2$ is large. Therefore, the misleading sample should satisfy the following property:

Property 1. *A misleading sample has a large inconsistency between $\mathcal{F}(\mathbf{x}^*)$ and $\mathcal{A}(\mathcal{H}(\mathbf{x}^*))$.*

Besides, based on the definition of the misleading sample, i.e., it is incorrectly classified with a high confidence, one logit of the misleading sample is close to 1. The misleading sample also needs to satisfy the following property:

Property 2. *The logit vectors of a misleading sample should be close to one-hot vectors.*

To this end, for each sample $\mathbf{x}_i$, we define a score $s_{mis}(\mathbf{x}_i)$ to characterize how likely this sample is a misleading one, where a large $s_{mis}(\mathbf{x}_i)$ means it is more like a misleading sample. According to Property 1, $s_{mis}(\mathbf{x}_i)$ should characterize the difference or similarity between $\mathcal{F}(\mathbf{x}^*)$ and $\mathcal{A}(\mathcal{H}(\mathbf{x}^*))$. Notice that $\mathcal{F}(\mathbf{x}_i)$ and $\mathcal{A}(\mathcal{H}(\mathbf{x}_i))$ are the logit vectors of $\mathbf{x}_i$ to all classes. In practice, one natural way to characterize the difference or similarity between two logit vectors is using the inner product $<\mathcal{F}(\mathbf{x}_i),\mathcal{A}(\mathcal{H}(\mathbf{x}_i))>$.

Considering Property 2, we wish the logit vectors $\mathcal{F}(\mathbf{x}_i)$ and $\mathcal{A}(\mathcal{H}(\mathbf{x}_i))$ are close to one-hot vectors. Since we often impose Softmax function on the last layer of $\mathcal{F}(\cdot)$ and $\mathcal{A}(\cdot)$ to obtain logits, we have $\|\mathcal{F}(\mathbf{x}_i)\|_1=1$ and $\|\mathcal{A}(\mathcal{H}(\mathbf{x}_i))\|_1=1$. When $\|\mathcal{F}(\mathbf{x}_i)\|_1=1$ and $\|\mathcal{A}(\mathcal{H}(\mathbf{x}_i))\|_1=1$, if we maximize $\|\mathcal{F}(\mathbf{x}_i)\|_2$ and $\|\mathcal{A}(\mathcal{H}(\mathbf{x}_i))\|_2$, it will lead to that one of the logits of $\mathcal{F}(\mathbf{x}_i)$ and $\mathcal{A}(\mathcal{H}(\mathbf{x}_i))$ is close to 1 and other logits are close to 0.

Therefore, according to Properties 1 and 2, a misleading sample should have a small $<\mathcal{F}(\mathbf{x}_i),\mathcal{A}(\mathcal{H}(\mathbf{x}_i))>$ and large $\|\mathcal{F}(\mathbf{x}_i)\|_2$ and $\|\mathcal{A}(\mathcal{H}(\mathbf{x}_i))\|_2$. To this end, we can use a simple cosine similarity to define $s_{mis}(\mathbf{x}_i)$:

$$\begin{aligned} s_{mis}(\mathbf{x}_i) &= 1-cos(\mathcal{F}(\mathbf{x}_i),\mathcal{A}(\mathcal{H}(\mathbf{x}_i))) \\ &= 1-\frac{<\mathcal{F}(\mathbf{x}_i),\mathcal{A}(\mathcal{H}(\mathbf{x}_i))>}{\|\mathcal{F}(\mathbf{x}_i)\|_2\|\mathcal{A}(\mathcal{H}(\mathbf{x}_i))\|_2} \end{aligned} \tag{2}$$

If the inner product $<\mathcal{F}(\mathbf{x}_i),\mathcal{A}(\mathcal{H}(\mathbf{x}_i))>$ is small, which means the difference between $\mathcal{F}(\mathbf{x}_i)$ and $\mathcal{A}(\mathcal{H}(\mathbf{x}_i))$ is large, $s_{mis}(\mathbf{x}_i)$ will be large and $\mathbf{x}_i$ will be more like a misleading sample. Besides, if $\|\mathcal{F}(\mathbf{x}_i)\|_2$ and $\|\mathcal{A}(\mathcal{H}(\mathbf{x}_i))\|_2$ is large, which means $\mathcal{F}(\mathbf{x}_i)$ and $\mathcal{A}(\mathcal{H}(\mathbf{x}_i))$ are close to one-hot vectors, $s_{mis}(\mathbf{x}_i)$ will also be large. Therefore, $s_{mis}(\mathbf{x}_i)$ is consistent with the two properties of misleading samples.

3.2 Boundary Sample Selection

Since the main contribution of this paper is to discover the misleading samples, for simplicity, we just borrow the traditional uncertainty-based method in [48] to identify the boundary samples. As illustrated in Figure 3, we apply a loss prediction module $\mathcal{P}$ [48] to predict the learning loss of the backbone network. In [48], it shows that if the learning loss $\mathcal{P}(\mathbf{x}_i)$ of a sample is large, this sample is more likely to be an uncertain sample. Therefore, we define the score of the boundary sample s_{bou} as the output of the loss prediction module $\mathcal{P}$, i.e., $s_{bou}(\mathbf{x}_i)=\mathcal{P}(\mathbf{x}_i)$. The structure and training of $\mathcal{P}$ is the same as in [48].

Here, we argue that although the method in [48] aims to predict the classification loss of samples, this method prefers to select the boundary samples instead of the misleading samples. When selecting samples for annotation, $\mathcal{P}$ evaluates the loss of samples in unlabeled data $\mathcal{U}$. On one hand, consider a misleading sample $\mathbf{x}^*$ belonging to the k-th class but appearing deep within the j-th class. Since $\mathbf{x}^*$ is unlabeled and deep within the j-th class, both the main classification network and the loss prediction module $\mathcal{P}$ may misclassify it into the j-th class, and thus $\mathcal{P}$ thinks it is an easy sample and predicts it with a lower learning loss. On the other hand, given a boundary sample $\mathbf{x}$, since the main classification network classifies it with lower confidence, $\mathcal{P}$ predicts it with a higher learning loss. Therefore, $\mathcal{P}$ prefers to select the boundary samples rather than the misleading samples. Our experimental results in Section 4.3 demonstrate this.

3.3 Adaptive Weighting Strategy

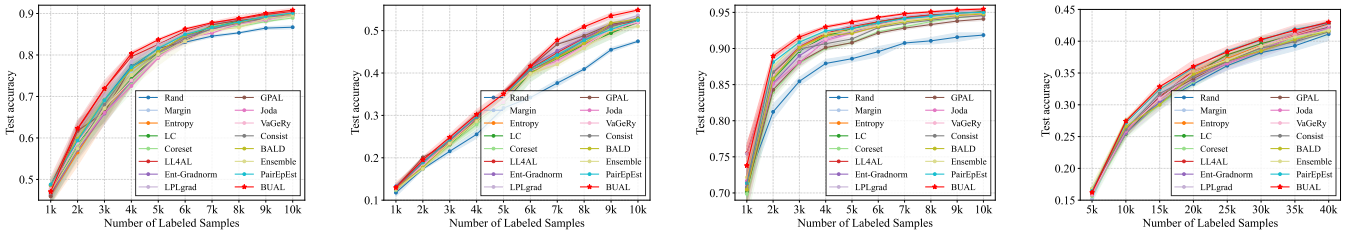
Equipped with s_{mis} and s_{bou}, we are ready to select uncertain samples for annotation. Notice that if we directly select samples from the entire unlabeled set $\mathcal{U}$, we may select similar samples for annotation because similar samples may have similar scores [3]. To address this issue, we adopt a simple technique from [3], which randomly samples a candidate subset $C\subseteq\mathcal{U}$ and then selects the samples in the candidate set C. To ensure a fair evaluation, we apply this subset selection strategy consistently across all methods.

Now, we need to select the final uncertain samples from C. Notice that s_{mis} and s_{bou} have different semantics and different scales, and thus it is meaningless to directly fuse them. When selecting samples, we only care about the order of the samples instead of the absolute uncertainty values of the samples. For example, given two samples $\mathbf{x}_i$ and $\mathbf{x}_j$, to decide which sample should be selected, we just need to know whether $\mathbf{x}_i$ is more uncertain than $\mathbf{x}_j$ instead of knowing the absolute uncertainty values of $\mathbf{x}_i$ and $\mathbf{x}_j$. To this end, we fuse the rank of s_{mis} and s_{bou} instead of directly fusing s_{mis} and s_{bou}. More specifically, we rank the samples in C in descending order of s_{mis}, denoted as $rank(s_{mis}(\mathbf{x}_i))$. Taking the sample $\mathbf{x}_j$ with the largest s_{mis} as an example, its rank $rank(s_{mis}(\mathbf{x}_j))=1$. For the boundary sample score, we similarly compute $rank(s_{bou}(\mathbf{x}_i))$ similar to $rank(s_{bou}(\mathbf{x}_i))$. Then we fuse the rank of misleading and boundary scores by computing its final score $s_{fin}(\mathbf{x}_i)$ as:

$$s_{fin}(\mathbf{x}_i)=\alpha\frac{rank(s_{mis}(\mathbf{x}_i))}{|C|}+(1-\alpha)\frac{rank(s_{bou}(\mathbf{x}_i))}{|C|}, \tag{3}$$

where α is an adaptive parameter to balance the ranks of misleading scores and boundary scores.

Notice that, at the beginning of training, the main network $\mathcal{F}(\cdot)$ and auxiliary network $\mathcal{A}(\cdot)$ may be weak, and their prediction may



(a) Cifar10 when the labeling budget is 10000. **(b) Cifar100 when the labeling budget is 10000.** **(c) SVHN when the labeling budget is 10000.** **(d) Tiny-ImageNet when the labeling budget is 40000.**

Figure 5: Classification results with different number of annotations on Cifar10, Cifar100, SVHN, and Tiny-ImageNet.

not be precise enough. Hence, it is difficult to discover the misleading samples accurately in the beginning. Our ablation studies (i.e., Figure 8) also demonstrate this. To this end, we set a small α in the beginning, and enlarge the weight α gradually. More specifically, suppose we have a total of T selection batches. In the t-th selection batch, we set $\alpha = \frac{1}{1+e^{m-t}}$ where $m = \frac{T}{2}$. It is easy to see that α increases with t, which means s_{mis} has an increasingly large weight in the process of training. In the middle of training, i.e., $t = \frac{T}{2}$, we have $\alpha = 0.5$, which means the misleading samples and boundary samples have the same weight. After that, misleading samples will have larger weights than boundary samples because the networks are strong enough to discover the difficult misleading samples. Here, m is a parameter to control when the misleading samples and boundary samples have the same weight. In our implementation, we fix it to $\frac{T}{2}$. In Appendix H, we show the effect of different m. At last, given a budget B of each selection batch, we select B samples from C with the smallest s_{fin}, i.e., the top-B uncertain samples, for annotation. The algorithm in Appendix E shows the whole process.

4 Experiment

4.1 Experimental Setup

We evaluate the performance of our method on benchmark datasets, including Cifar10 [24], Cifar100 [24], SVHN [33], and Tiny-ImageNet [25]. The details of these datasets are introduced in Appendix F.

We compare our approach with 14 active learning methods, including **Margin** [36], **Entropy** [28], **LC** [44], **Coreset** [38], **BALD** [11], **Ensemble** [3], **LL4AL** [48], **Consist** [12], **Ent-GradNorm** [47], **LPLgrad** [13], **GPAL** [1], **Joda** [37], **VaGeRy** [5] and **PairEpEst** [4]. We also compare with the random selection as a baseline, denoted as **Random**. To ensure reliable results, we repeat the experiments 5 times and report the average results and the standard deviation.

Following [13, 47, 48], we use ResNet-18 as the classification model for all the experiments. We adopt the wavelet package sym17 [26] to implement WWT. The parameter settings are the same as [13], whose details are introduced in Appendix F.

4.2 Experimental Results

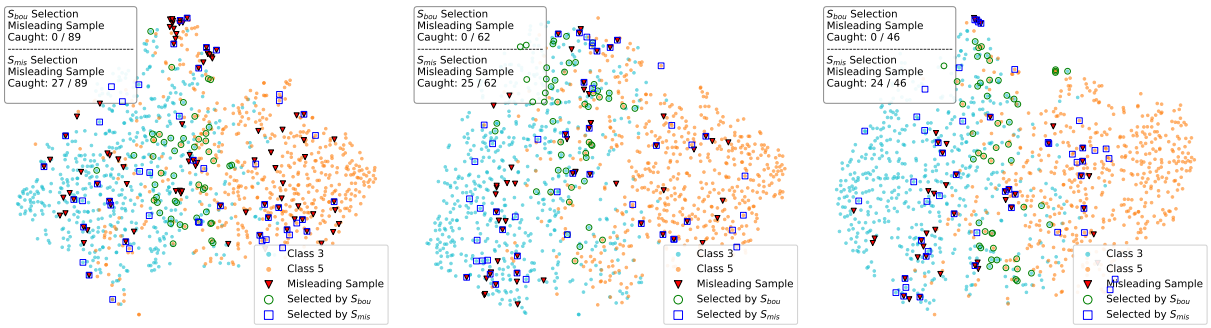
Following [13, 48], we set the number of selection batches as $T = 10$, and in each selection batch, the selection budget is 1000 samples for Cifar10, Cifar100, and SVHN. Besides, on Tiny-ImageNet, which contains many more classes, following [37], we set $T = 8$, and the budget of each selection batch is 5000 samples. The average test accuracy and standard deviation of all methods are shown in Figure 5. Figure 5 shows that our method outperforms other active learning methods most of the time on all datasets. Moreover, our method can achieve better performance with fewer annotations. For example, in Cifar10, when we annotate 4000 samples (8% samples), our method achieves an average accuracy of 80.33%, surpassing the second-best method by 0.75%. In Cifar100, when we annotate 10000 samples (20% of samples), ours achieves an accuracy of 54.86%, while the second-best one only achieves 53.60%. In SVHN, when annotating 2000 samples (2.73% of samples), our accuracy is 88.95%, and the second-best one is 88.12%. In Tiny-ImageNet, when we annotate 20000 samples (20% of samples), ours achieves an accuracy of 36.03%, while the second-best one achieves 35.86%. These results well demonstrate the superiority of our active learning method.

Notice that we use ResNet-18 as the classification model for a fair comparison, but our method can also be applied to other backbones. In Appendix G, we show the results with a more popular backbone, ResNet-50, for image classification. Besides, we show the efficiency results in Appendix J.

4.3 Effectiveness of Identifying Misleading Samples

Since one of our main contributions is to discover the misleading samples, in this subsection, we show some results to verify whether our method can discover them. Firstly, Figure 6 illustrates the t-SNE visualization of our BUAL on Cifar10. The figure shows plots for the "cat" and "dog" classes obtained from the 1st, 4th, and 7th selection batches, respectively. Red triangles denote the real misleading samples, which means the model misclassifies them with confidence greater than 0.98. Blue squares indicate samples selected by our misleading sample discovering method (Section 3.1.2), and green circles represent samples chosen by the traditional learning loss based method (i.e., s_{bou} or LL4AL [48], introduced in Section 3.2). We can see that, compared with LL4AL, our method can discover more misleading samples. For example, when in the 7-th selection batch, our method can identify 24 from 46 real misleading samples, but LL4AL finds none of them. Besides, as shown in these figures, LL4AL prefer to discover the boundary samples. That is why we apply LL4AL [48] to identify the boundary samples in our method.

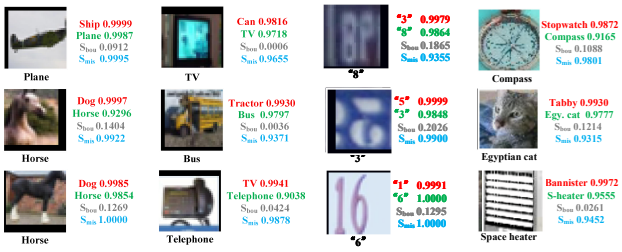
Secondly, Figure 7 presents some examples of misleading samples from all datasets. The red and green fonts denote the predicted



(a) Selection batch 1 **(b) Selection batch 4** **(c) Selection batch 7**

Figure 6: t-SNE visualization of sample selection on the cat and dog classes in Cifar10.

classes and confidence scores produced by the main network $\mathcal{F}(\cdot)$ and the auxiliary network $\mathcal{A}(\cdot)$, respectively. Gray and blue fonts denote the boundary score s_{bou} and the misleading score s_{mis}, respectively. For these misleading samples, our method often gives a higher misleading score, whereas LL4AL [48] often gives a lower boundary score, which is consistent with our motivation of discriminating between the two kinds of uncertain samples.



(a) Cifar10 **(b) Cifar100** **(c) SVHN** **(d) Tiny-ImageNet**

Figure 7: Examples of misleading samples with their classification results, confidence, boundary score, and misleading score across all datasets.

Lastly, to quantitatively evaluate the capability of discovering misleading samples, we report the number of successfully selected misleading samples across different active learning methods on the Cifar10 dataset in Table 1. To ensure the reliability of our results, this experiment is also repeated 5 times, and the average numbers are reported. Specifically, in each selection batch, every method evaluates an unlabeled subset of 10000 samples and selects the top 1000 samples with the highest scores for annotation. In Table 1, "a/b" means that there are totally b real misleading samples, and the method discovers a misleading samples of them (i.e., recall of misleading samples). As shown in Table 1, traditional uncertainty-based methods (i.e., Margin, LC, Entropy, Ent-GradNorm, and LL4AL) completely fail to identify these samples, capturing almost none of the misleading samples. This is because these methods strictly query low-confidence samples, and inherently ignore the high-confidence misclassification samples. For the random sampling baseline, since the querying budget (1,000) is 10% of the subset size (10,000), it naturally captures approximately 10% of the misleading samples simply by statistical chance (e.g., capturing 150 out of 1430 in batch 1). The inconsistency-based methods (e.g., Consist, BALD, Ensemble and PairEpEst) can discover a little more misleading samples than Rand. However, our proposed misleading score s_{mis} consistently discovers significantly more misleading samples than Rand and other inconsistency-based methods across all selection batches. For instance, our method successfully captures an average of 274 out of 1430 real misleading samples in the first batch, and maintains a strong discovery capability in later stages (e.g., capturing 53 out of 162 in batch 9). This quantitative evidence shows that our strategy can more effectively identify the high-confidence yet incorrect samples that conventional active learning methods may ignore.

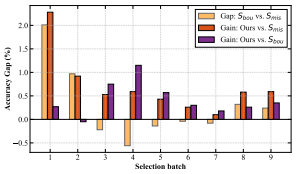
Table 1: Number of discovered misleading samples and real misleading samples across different active learning methods in each selection batch on the Cifar10 dataset (# of discovered misleading samples / # of real misleading samples).

Method		Selection batch				
		1	3	5	7	9
Rand		150/1430	64/598	42/341	24/226	15/162
Uncertainty	Margin	0/1430	0/598	0/341	0/226	0/162
	LC	0/1430	0/598	0/341	0/226	0/162
	Entropy	0/1430	0/598	0/341	0/226	0/162
	EntG	0/1430	0/598	0/341	0/226	0/162
	s_{bou}(LL4AL)	5/1430	0/598	0/341	0/226	0/162
Inconsistency	Consist	100/1430	48/598	54/341	47/226	38/162
	BALD	171/1430	74/598	40/341	24/226	15/162
	Ensemble	125/1430	64/598	34/341	27/226	22/162
	PairEpEst	179/1430	79/598	43/341	32/226	24/162
s_{mis}		**274**/1430	**129**/598	**105**/341	**75**/226	**53**/162

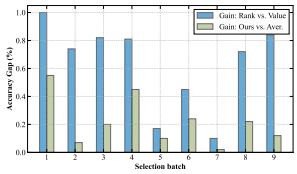
4.4 Ablation Study

In this subsection, we conduct comprehensive ablation studies to evaluate the main components and design choices of our method. Specifically, we investigate: the respective effects of the boundary score s_{bou} and the misleading score s_{mis}, as well as their complementarity; the effectiveness of the proposed adaptive weighting and rank-based fusion strategies; whether our bi-modal method outperforms conventional inconsistency-based active learning strategies; the effect of cosine similarity in Eq.(2) compared with other conventional inconsistency measures. We further analyze the effectiveness of the weighted wavelet transform (WWT) in Appendix I.

We first analyze the individual and complementary effects of the boundary and misleading scores. Figure 8a presents the results on Cifar10. The yellow bars represent the performance gap between only using s_{bou} and only using s_{mis} in each selection batch. We observe that in early stages (i.e., the first and second selection batches), boundary samples are much more useful than misleading samples; and in the next several batches, misleading samples play more important roles. This may be because, in early stages, misleading samples are too difficult to discover, and boundary samples are easy to identify, which can be used to effectively train the model. This motivates our adaptive weighting strategy that imposes small weights on misleading samples in early stages and gradually increases them (see Section 3.3). The orange and purple bars show the performance gain of our method compared to only selecting the misleading samples and the boundary samples, respectively. It can be seen that our method of selecting both samples is better than selecting a single kind of sample. It demonstrates that the misleading samples proposed in this paper are indeed useful to AL.



(a) Comparison of the performance gaps and gains among only using s_{bou}, only using s_{mis}, and using both scores.



(b) Comparison of the gain of our adaptive weighting strategy compared with the average weighting and value fusion methods.

Figure 8: Ablation study results on the Cifar10 dataset.

Figure 8b shows the effects of our adaptive weighting strategy. The green bars show the performance gap between our method and the method of average weighting misleading and boundary samples (i.e., fixing $\alpha = 0.5$ in Eq.(3)). The adaptive method always outperforms the average method, demonstrating the effectiveness of the adaptive weighting strategy introduced in Section 3.3. The blue bars show the gains of our rank fusion method compared to the value fusion method. We can see that the rank fusion method always outperforms the value fusion method. As discussed in Section 3.3, the two scores possess entirely different semantics and scales. Consequently, directly fusing their absolute raw values may lead to suboptimal and unstable sample selection.

We next evaluate whether the proposed misleading-sample score provides a more effective inconsistency criterion than existing inconsistency-based active learning strategies. We conduct ablation studies to compare with several conventional inconsistency-based strategies, including Consist [12], BALD [11], Ensemble [3], and PairEpEst [4], i.e., we replace our selection strategy with these methods. The accuracy is shown in Figure 9. Our method achieves the best accuracy compared to these strategies most of the time. Moreover, Table 1 also shows the performance of these strategies to discover misleading samples (recall of misleading samples). We can see that our strategy can find more misleading samples compared to these inconsistency-based strategies.

In Eq.(2), we utilize cosine similarity to quantify the cross-modal inconsistency between the main model and the auxiliary model,

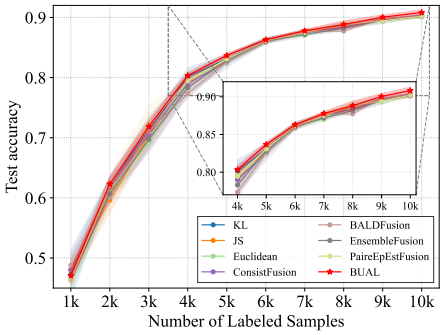


Figure 9: Ablation study of the inconsistency active learning strategies and different inconsistency measures on Cifar10.

we evaluate this choice by comparing it against three conventional distribution-matching metrics: Kullback-Leibler (KL) divergence, Jensen-Shannon (JS) divergence and Euclidean distance. Figure 9 shows the results. Cosine similarity consistently yields superior performance across all active learning batches on Cifar10 compared to these divergences. Although KL divergence, JS divergence, and Euclidean distance are widely used to measure the distance between two probability distributions, they are not optimal for our specific task. As analyzed in Section 3.1.2, misleading samples possess two distinct properties: (1) High cross-modal inconsistency, meaning the main and auxiliary models yield diverging predictions; and (2) High prediction confidence, meaning the logit vectors approximate one-hot distributions. KL and JS divergences only focus on distribution distance (i.e., Property 1), while ignoring Property 2. Different from KL and JS divergences, our cosine similarity elegantly captures both properties. Minimizing the inner product captures the directional divergence (inconsistency) between the two modalities, while maximizing the denominator naturally prioritizes vectors with larger L2 norms (high confidence). Hence, cosine similarity can simultaneously characterize both properties.

5 Conclusion

This paper investigated a long-overlooked yet critical issue in uncertainty-based active learning: there exists a type of uncertain samples that even the model itself does not know is uncertain, which we called "misleading samples". We theoretically analyzed the generation and properties of the misleading samples. Based on the analysis, we proposed a simple yet effective method to identify the misleading samples. We designed an adaptive weighted strategy to integrate the misleading samples and the traditional boundary samples, to select the uncertain samples more comprehensively. Extensive experiments on the image classification task show the effectiveness and superiority of our proposed method. Since this is the first work to explore the misleading samples in active learning, we design a simple method to identify the misleading samples. As shown in Table 1, there is still plenty of room for improvement, and it is worthy of further research in the future. Besides, this work only focuses on the uncertainty, and in the future, we will plug it into hybrid active learning methods that consider both the uncertainty and representativeness.

6 Acknowledgments

This work is supported by the National Natural Science Foundation of China grants 62576002, 62176001, and the Natural Science Project of Anhui Provincial Education Department grant 2023AH030004. This work is also supported by the Anhui Provincial Key Research and Development Program under Grant 202304a05020047.

References

[1] Yunpyo An, Suyeong Park, and Kwang In Kim. 2024. Active learning guided by efficient surrogate learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 10874–10881.

[2] Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. 2019. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671* (2019).

[3] William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. 2018. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 9368–9377.

[4] Lucas Berry and David Meger. 2026. Epistemic Uncertainty Estimation in Regression Ensemble Models with Pairwise Epistemic Estimators. *Advances in Neural Information Processing Systems* 38 (2026), 64503–64536.

[5] Weiguo Chen, Changjian Wang, Shijun Li, Kele Xu, Yanru Bai, Wei Chen, and Shanshan Li. 2025. Debiased Active Learning with Variational Gradient Rectifier. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 15884–15894.

[6] Zhuoxiao Chen, Yadan Luo, Zixin Wang, Zijian Wang, and Zi Huang. 2025. Open-CRB: Toward Open World Active Learning for 3D Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 47, 10 (2025), 8336–8350.

[7] David A Cohn, Zoubin Ghahramani, and Michael I Jordan. 1996. Active learning with statistical models. *Journal of artificial intelligence research* 4 (1996), 129–145.

[8] Sayna Ebrahimi, Anna Rohrbach, and Trevor Darrell. 2017. Gradient-free policy architecture search and adaptation. In *Conference on Robot Learning*. PMLR, 505–514.

[9] Ehsan Elhamifar, Guillermo Sapiro, Allen Yang, and S Shankar Sasrty. 2013. A convex optimization framework for active learning. In *Proceedings of the IEEE international conference on computer vision*. 209–216.

[10] Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*. PMLR, 1050–1059.

[11] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. 2017. Deep bayesian active learning with image data. In *International conference on machine learning*. PMLR, 1183–1192.

[12] Mingfei Gao, Zizhao Zhang, Guo Yu, Sercan Ö Arık, Larry S Davis, and Tomas Pfister. 2020. Consistency-based semi-supervised active learning: Towards minimizing labeling cost. In *European Conference on Computer Vision*. Springer, 510–526.

[13] Shreen Gul, Mohamed Elmahallawy, Sanjay Madria, and Ardhendu Tripathy. 2024. LPLgrad: Optimizing Active Learning Through Gradient Norm Sample Selection and Auxiliary Model Training. In *2024 IEEE International Conference on Big Data (BigData)*. IEEE, 900–909.

[14] Yuhong Guo. 2010. Active instance sampling via matrix partition. *Advances in neural information processing systems* 23 (2010).

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[16] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. 2011. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745* (2011).

[17] Zitong Huang, Ze Chen, Yuanze Li, Bowen Dong, Erjin Zhou, Yong Liu, Rick Siow Mong Goh, Chun-Mei Feng, and Wangmeng Zuo. 2024. Class balance matters to active class-incremental learning. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 9445–9454.

[18] Michael Kampffmeyer, Arnt-Borre Salberg, and Robert Jenssen. 2016. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 1–9.

[19] Ashish Kapoor, Kristen Grauman, Raquel Urtasun, and Trevor Darrell. 2007. Active learning with gaussian processes for object categorization. In *2007 IEEE 11th international conference on computer vision*. IEEE, 1–8.

[20] Kwanyoung Kim, Dongwon Park, Kwang In Kim, and Se Young Chun. 2021. Task-aware variational adversarial active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8166–8175.

[21] Andreas Kirsch, Joost Van Amersfoort, and Yarin Gal. 2019. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *Advances in neural information processing systems* 32 (2019).

[22] Ksenia Konyushkova, Raphael Sznitman, and Pascal Fua. 2017. Learning active learning from data. *Advances in neural information processing systems* 30 (2017).

[23] Christine Körner and Stefan Wrobel. 2006. Multi-class ensemble-based active learning. In *European conference on machine learning*. Springer, 687–694.

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

[25] Yann Le, Xuan Yang, et al. 2015. Tiny imagenet visual recognition challenge. *CS 231N* 7, 7 (2015), 3.

[26] Gregory Lee, Ralf Gommers, Filip Waselewski, Kai Wohlfahrt, and Aaron O'Leary. 2019. PyWavelets: A Python package for wavelet analysis. *Journal of Open Source Software* 4, 36 (2019), 1237.

[27] Adrian S Lewis and G Knowles. 1992. Image compression using the 2-D wavelet transform. *IEEE Transactions on image Processing* 1, 2 (1992), 244–250.

[28] Xin Li and Yuhong Guo. 2013. Adaptive active learning for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 859–866.

[29] Xinyu Li, Wenqing Ye, Yueyi Zhang, and Xiaoyan Sun. 2024. Grace: Gradient-based active learning with curriculum enhancement for multimodal sentiment analysis. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 5702–5711.

[30] Oisin Mac Aodha, Neill DF Campbell, Jan Kautz, and Gabriel J Brostow. 2014. Hierarchical subquery evaluation for active learning on a graph. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 564–571.

[31] Waleed A Mahmoud, Ahmed S Hadi, and Talib M Jawad. 2012. Development of a 2-D Wavelet Transform based on Kronecker Product. *Al-Nahrain Journal of Science* 15, 4 (2012), 208–213.

[32] Jishnu Mukhoti, Andreas Kirsch, Joost Van Amersfoort, Philip HS Torr, and Yarin Gal. 2023. Deep deterministic uncertainty: A new simple baseline. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 24384–24394.

[33] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. 2011. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, Vol. 2011. Granada, 7.

[34] Hieu T Nguyen and Arnold Smeulders. 2004. Active learning using pre-clustering. In *Proceedings of the twenty-first international conference on Machine learning*. 79.

[35] Anant Raj and Francis Bach. 2022. Convergence of Uncertainty Sampling for Active Learning. In *Proceedings of the 39th International Conference on Machine Learning*. 18310–18331.

[36] Dan Roth and Kevin Small. 2006. Margin-based active learning for structured output spaces. In *European conference on machine learning*. Springer, 413–424.

[37] Sebastian Schmidt, Leonard Schenk, Leo Schwinn, and Stephan Günnemann. 2025. Joint out-of-distribution filtering and data discovery active learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 25677–25687.

[38] Ozan Sener and Silvio Savarese. 2017. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489* (2017).

[39] Feifei Shao, Yawei Luo, Ping Liu, Jie Chen, Yi Yang, Yulei Lu, and Jun Xiao. 2022. Active learning for point cloud semantic segmentation via spatial-structural diversity reasoning. In *Proceedings of the 30th ACM International Conference on Multimedia*. 2575–2585.

[40] Meng Shen, Yizheng Huang, Jianxiong Yin, Heqing Zou, Deepu Rajan, and Simon See. 2023. Towards balanced active learning for multimodal classification. In *Proceedings of the 31st ACM International Conference on Multimedia*. 3434–3445.

[41] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. 2019. Variational adversarial active learning. In *Proceedings of the IEEE/CVF international conference on computer vision*. 5972–5981.

[42] Ying-Peng Tang and Sheng-Jun Huang. 2022. Active learning for multiple target models. *Advances in Neural Information Processing Systems* 35 (2022), 38424–38435.

[43] Jinyu Tian, Jiantao Zhou, Yuanman Li, and Jia Duan. 2021. Detecting adversarial examples from sensitivity inconsistency of spatial-transform domain. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 9877–9885.

[44] Dan Wang and Yi Shang. 2014. A new active labeling method for deep learning. In *2014 International joint conference on neural networks (IJCNN)*. IEEE, 112–119.

[45] Hanmo Wang, Liang Du, Peng Zhou, Lei Shi, and Yi-Dong Shen. 2015. Convex Batch Mode Active Sampling via α-Relative Pearson Divergence. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, Blai Bonet and Sven Koenig (Eds.). AAAI Press, 3045–3051.

[46] Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P Xing. 2020. High-frequency component helps explain the generalization of convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8684–8694.

[47] Tianyang Wang, Xingjian Li, Pengkun Yang, Guosheng Hu, Xiangrui Zeng, Siyu Huang, Cheng-Zhong Xu, and Min Xu. 2022. Boosting active learning via improving test performance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 8566–8574.

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

[49] Beichen Zhang, Liang Li, Shijie Yang, Shuhui Wang, Zheng-Jun Zha, and Qingming Huang. 2020. State-relabeling adversarial active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8756–8765.

[50] Helin Zhao, Wei Chen, and Peng Zhou. 2024. Active Deep Multi-view Clustering. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024*. ijcai.org, 5554–5562.

[51] Peng Zhou, Bicheng Sun, Xinwang Liu, Liang Du, and Xuejun Li. 2024. Active Clustering Ensemble With Self-Paced Learning. *IEEE Trans. Neural Networks Learn. Syst.* 35, 9 (2024), 12186–12200.