FastAL: Fast Evaluation Module for Efficient Dynamic Deep Active Learning Using Broad Learning System

Shuzhou Sun, Huali Xu, Yan Li, Ping Li, *Member, IEEE*, Bin Sheng, *Member, IEEE*, and Xiao Lin

***Abstract*— State-of-the-art Active Learning (AL) methods often encounter challenges associated with a hysteretic learning process and an expensive data sampling mechanism. The former implies that data selection in the $(i+1)$-th round is solely based on the learned model's results in the i-th round. The latter involves using model inference to calculate data value (e.g., uncertainty estimation based on model inference), which can be cumbersome, particularly when working with large datasets or Deep Neural Networks (DNNs). To address these challenges, we propose FastAL, an efficient and dynamic deep AL framework. Our approach includes an efficient method for calculating data value from the frequency domain perspective, generating multiple candidates. Then, we introduce the Fast Evaluation Module, which directly calculates each candidate's contribution to future model training and selects the best options. In addition, current AL methods, particularly those based on uncertainty, are susceptible to data bias, which implies that selected data may not represent the original unlabeled data adequately. To alleviate this issue, we propose the De-similar Module, which removes partially similar data. The above three modules are model-agnostic and thus can be seamlessly integrated into any Active Learning framework. We conducted rigorous experiments on various benchmark datasets to validate our approach's effectiveness. Our results demonstrate that FastAL outperforms other state-of-the-art methods by a significant margin, including those based on uncertainty, diversity, and expected model change.**

Manuscript received 13 March 2023; revised 12 June 2023; accepted 17 June 2023. Date of publication 28 June 2023; date of current version 6 February 2024. This article was recommended by Associate Editor X. Cao. *(Shuzhou Sun and Huali Xu contributed equally to this work.) (Corresponding author: Xiao Lin.)*

Shuzhou Sun is with the College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai 200234, China, and also with the Center for Machine Vision and Signal Analysis, University of Oulu, 8000 Oulu, Finland (e-mail: Shuzhou.Sun@oulu.fi).

Huali Xu is with the Center for Machine Vision and Signal Analysis, University of Oulu, 8000 Oulu, Finland (e-mail: Huali.Xu@oulu.fi).

Yan Li is with the College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai 200234, China (e-mail: yanli@shnu.edu.cn).

Ping Li is with the Department of Computing and the School of Design, The Hong Kong Polytechnic University, Hong Kong (e-mail: p.li@polyu.edu.hk).

Bin Sheng is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: shengbin@sjtu.edu.cn).

Xiao Lin is with the College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai 200234, China, also with the Shanghai Engineering Research Center of Intelligent Education and Bigdata, Shanghai 200240, China, and also with the Research Base of Online Education for Shanghai Middle and Primary Schools, Shanghai 200234, China (e-mail: lin6008@shnu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2023.3288134.

Digital Object Identifier 10.1109/TCSVT.2023.3288134

***Index Terms*— Active learning, broad learning system (BLS), neural network, data bias, deep learning.**

I. Introduction

THANKS to the immense amounts of labeled data, Deep Neural Networks (DNNs) have achieved remarkable success in various areas such as classification [1], [2], object detection [3], [4], and semantic segmentation [5], [6]. However, in fields that require professional expertise (e.g., medical images), the labeling of data for network training still incurs prohibitive costs [7], [8], [9], [10]. To overcome this problem, Active Learning (AL) samples the most valuable data from the unlabeled pool for model training. AL methods can thus mitigate the expense of labeling by discarding low or even no-value unlabeled data. AL has been proposed and applied for many years, with typical methods including uncertainty-based methods [11], [12], [13], [14], [15], diversity-based methods [16], [17], expected model change AL methods [18], [19], [20], etc. Albeit its prosperity, those AL methods still have the following challenges.

First, AL methods currently follow the hysteretic paradigm, where unlabeled samples are selected based on the model learned in the previous round. For example, uncertainty-based AL methods [11], [12], [21] utilize the model learned in the previous round to calculate the uncertainty of unlabeled data for the current round. Likewise, expected model change AL methods [18], [19], [20] use processed unlabeled data (e.g., noise) as inputs in the previous round to observe the changes in the model outputs. While these methods can leverage the learning achievements of the previous round, it remains a significant challenge to ensure that the training data of the previous rounds cover/represent the entire unlabeled data pool. Consequently, selecting data based on training achievements from the previous round does not guarantee continuous contribution to future model training. Our proposed **Fast Evaluation Module** addresses this challenge by directly computing the data's contribution to future model learning. Fig. 1 illustrates the difference between the conventional AL framework and our proposed FastAL framework. More specifically, the Fast Evaluation Module can directly calculate the future contribution of the candidates selected by the learned model. The Fast Evaluation Module is based on Broad Learning System [22], an efficient model applicable to classification and regression

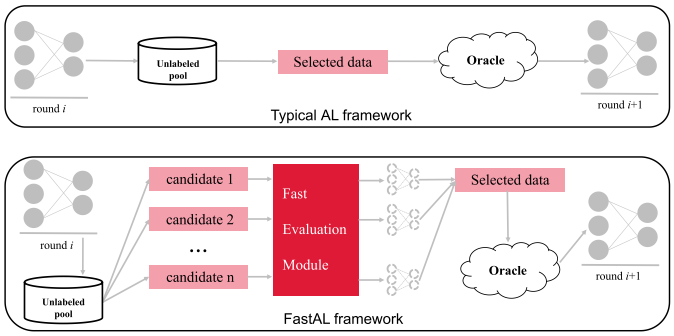


Fig. 1. The typical AL framework and our proposed FastAL framework. The typical AL framework obtains the selected data based on the learned model in round i. While our proposed FastAL framework first gets the multiple candidates based on the learned model in round i and then uses the Fast Evaluation Module to select the one that contributes most to future round learning.

problems [23]. Thus, the Fast Evaluation Module can maintain low costs, even with large-scale unlabeled data and Deep Neural Networks (DNNs).

Secondly, the process of data selection in Active Learning (AL), particularly in deep AL approaches, is often influenced by the model being used. For instance, Loss Prediction Module (LPM) [24] treats unlabeled data as part of model training to predict the target losses of unlabeled inputs. In Localization-Aware AL [21], uncertainty is calculated based on the tightness and stability of localization. However, the learned model must provide intermediate predictions. Therefore, the model-dependent methods necessitate redesigning when learning new models or tasks, which can be a time-consuming and labor-intensive process. To address this issue, we propose a **model-agnostic value calculation** method to calculate the training contribution of unlabeled data directly without relying on the learned model. In particular, the proposed model-agnostic value calculation method calculates the value of the unlabeled data from the perspective of the frequency domain, which is inspired by the fact that the frequency domain information of the data is consistent with the information perceived by the DNNs. For instance, Discrete Cosine Transform (DCT) [25] reveals that in addition to visual data itself, incorporating frequency domain information as input to neural networks enhances the model's representational capacity. High-Order Relation module (HOR) [26] highlighted the consistency of frequency domain information with RGB domain features and integrated them to improve camouflaged object detection. More importantly, our model-agnostic method for value calculation is more efficient when compared to model-dependent methods.

Finally, traditional Active Learning (AL) methods, particularly those that rely on uncertainty-based techniques [13], [14], [15], [27], are susceptible to data bias. This issue arises primarily from the imbalanced distribution and diverse levels of difficulty of different categories within the training data. As a result, the model learned in the previous round is inevitably biased toward some categories while underperforming in others. For example, if the previously learned model fails to capture essential features of categories such as *cows*, *buses*, or *plants*, it may result in high uncertainty for the unlabeled data belonging to these categories. However, the biased AL methods primarily emphasize sampling from a few categories with high levels of uncertainty while neglecting the majority of other classifications, hence leading to an imbalance in the sampling process. Unbalanced sampling is inadequate in representing the original dataset adequately, consequently leading to data bias problems. Despite extensive research efforts [11], [12], [21], [28] focused on this issue, most methods require significant additional training costs and involve designing complex loss functions artificially. In this paper, We propose a simple but effective method named the **De-similar Module** to mitigate data bias. The De-similar Module calculates the similarity between selected data and then removes those with a high degree of similarity.

In summary, our work makes the following main contributions:

- We thoroughly analyze the typical Active Learning (AL) frameworks that encounter challenges associated with a hysteretic learning process and an expensive data sampling process.
- We present a novel, efficient, and dynamic deep Active Learning (AL) framework, named FastAL, which is composed of three key modules. The first module is the **model-agnostic value calculation module** that utilizes frequency domain information to quickly assess the value of unlabeled data, irrespective of the training outcomes of the previous AL round. The second module is the **De-similar Module** which calculates the degree of similarity between the selected data and eliminates redundant data based on the amount of information present in the data, thereby mitigating data bias problems. The third module is the **Fast Evaluation Module**, which enables the direct evaluation of the contribution of selected unlabeled data to the subsequent model training. To expedite this process, we propose the use of the Broad Learning System (BLS) as an evaluation procedure. Notably, the three modules in FastAL are model-agnostic and thus can be used in any AL framework.
- Comprehensive experiments were conducted on various network backbones. The results show that our FastAL achieves state-of-the-art performance on four popular benchmarks in terms of two metrics, where one is the prevalent metric in the active learning literature, and another is a novel metric introduced in this paper.

II. Related Work

A. Deep Active Learning

Active learning has attracted widespread attention in recent years due to its impressive performance in saving labeling costs. The typical AL methods can be roughly classified as the uncertainty-based AL methods [11], [12], [13], [14], [15], the diversity-based AL methods [16], [17], and the expected model change AL methods [18], [19], [20].

The uncertainty-based AL methods calculate the uncertainty of unlabeled data based on the learned model in the previous round. In addition to traditional methods such

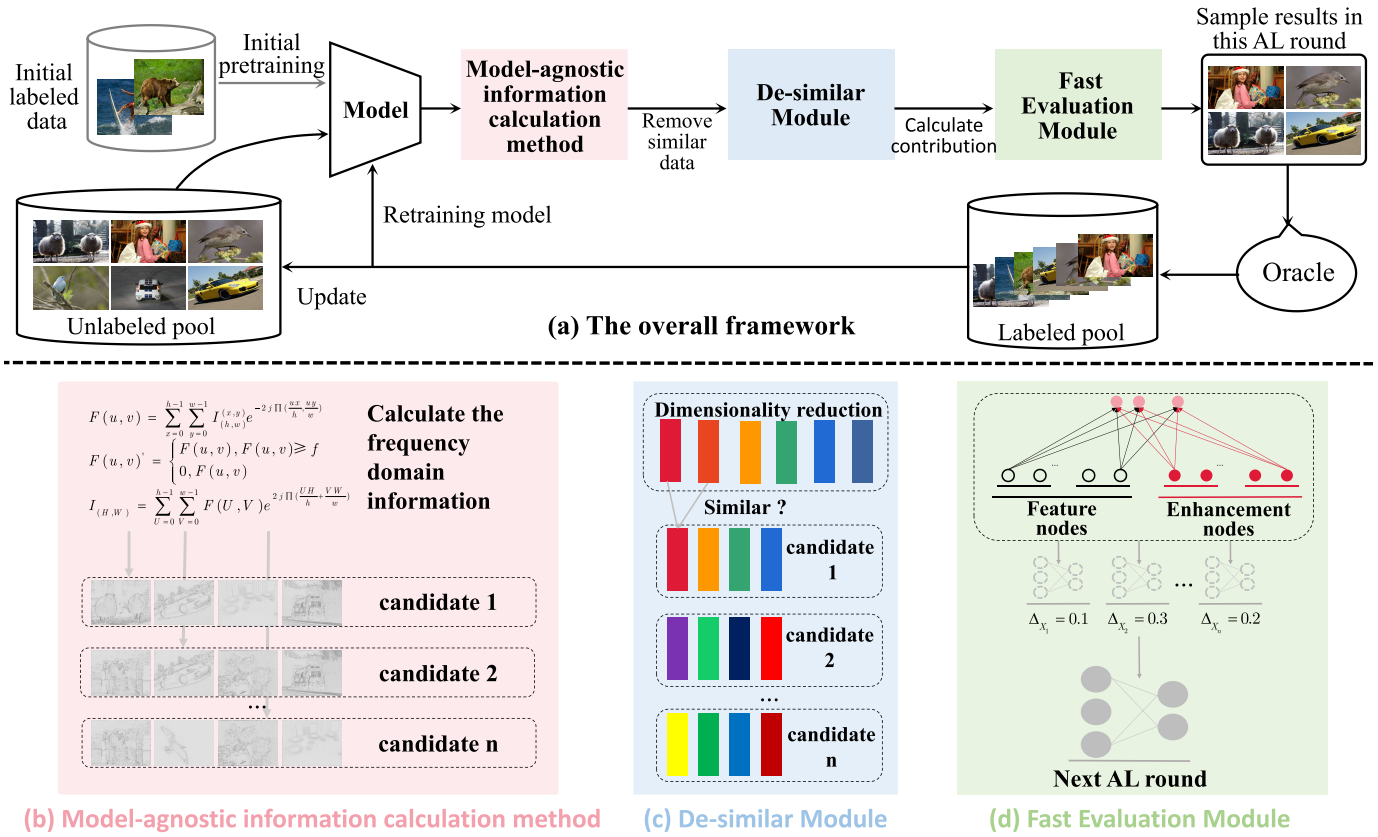


Fig. 2. The overall pipeline of FastAL. FastAL first abandons traditional, high-cost data sampling mechanisms, such as uncertainty estimation based on model inference, and instead uses a model-agnostic value calculation method to assess the value of unlabeled data and select multiple candidates. Then, the proposed De-similar Module removes redundant data from the candidate pool to mitigate data bias. Finally, the Fast Evaluation Module selects the candidate that best contributes to future Active Learning rounds, overcoming challenges induced by the hysteretic learning process.

as Least Confidence Sampling [13], Margin Sampling [14], Entropy Sampling [15], among others, some recent efforts fall into this category. For example, Sparse Modeling Active Learning (SMAL) [11] uses the sparse linear combination to represent the uncertainty of unlabeled data with Gaussian kernels. However, the sparse representation is challenging to guarantee stability when facing large-scale data. Ensembles-based Active Learning (ENS) [12] uses an ensemble network to calculate data uncertainty. Obviously, ENS introduces a lot of additional computational costs, especially when facing large-scale unlabeled data and DNNs.

The diversity-based AL methods select a batch sample from the perspective of data features to cover the entire unlabeled data. Fisher Kernel Self Supervision (FKSS) [16] uses the proposed feature density matching methods to predict the contribution of unlabeled data to future training. The Core-set approach [17] defines the problem of AL as core-set selection and chooses a set of points such that a model learned over the selected subset is competitive for the remaining data points. Integer Programming Approach (IPA) [29] minimizes the discrete Wasserstein distance in feature space from the unlabeled pool to select the core set. ALFA-Mix [30] identifies unlabeled samples with sufficiently-distinct features by seeking inconsistencies in predictions resulting from interventions on their representations. Although the above methods have been shown to be effective for simple and low-scale features, they will introduce unaffordable costs. Besides, the feature engineering establishment is unstable when facing more complex and large-scale features.

The expected model change AL methods process the unlabeled data (e.g., adding noise) and then feed them into the learned model of the previous round to observe the changes in the model outputs. Settles et al. [18] estimate the value of unlabeled data by measuring the changes in model parameters but ignore the underlying data distribution. To address this issue, Freytag et al. [19] directly calculate the expected change of model predictions and marginalize the unknown label. Further, Käding et al. [20] leverage the approximated gradients of the loss function to enable the Gaussian process regression AL methods can be used for the deep convolutional neural network models. However, the expected model change AL methods introduce the cost of unlabeled data processing and model reasoning.

In this paper, we propose a novel AL framework FastAL. FastAL starts with a model-agnostic value calculation method coupled with a De-similar Module to obtain multiple candidates. Then, our proposed Fast Evaluation Module selects the candidate who contribute the most to future model learning.

B. Broad Learning System

Although the DNNs have achieved impressive performance on multiple tasks, the training process requires unbearable computing resources and time due to a large

number of parameters and complex hand-designed structures [22], [23], [31], [32], [33], [34]. In addition, the scalability of DNNs is limited because it is difficult for the learned model to continue to learn new knowledge in the face of new training data.

To address those troubles, Chen et al. design the Broad Learning System (BLS) [22], a training-efficient and scalability-friendly network structure for classification and regression problems. Meanwhile, Chen et al. [23] also proved that the BLS has the same universal approximation capability as the single-layer feedforward network (SLFN) [35]. The vanilla BLS [22] includes the feature nodes and the enhancement nodes. The former transferred the original inputs and placed them as "mapped features," and the latter expanded the structure in the broad sense. When faced with new data in the open world, the BLS can achieve "learning new data without forgetting old knowledge" by expanding the feature and enhancement nodes. Note that this process is to learn only new data on the basis of maintaining the original learning achievements. Also, the BLS showed a huge advantage in training efficiency, i.e., the training times of MLP, CNN, and BLS on the MNIST [36] are 21468.12s, 21793.93s, and 29.9157s respectively. Inspired by the above facts, we introduce the BLS to the AL framework and propose the Fast Evaluation Module based on it. The Fast Evaluation Module can efficiently select the one that contributes the most to future model learning from multiple candidates.

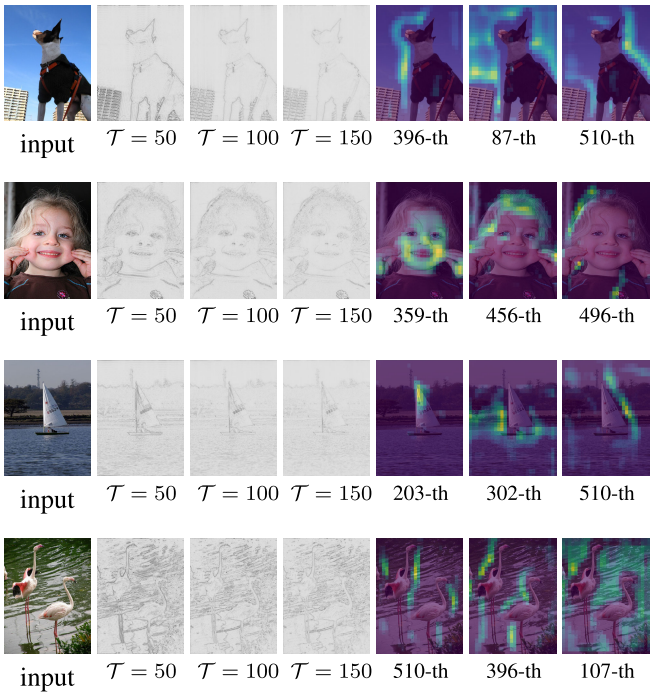


Fig. 3. The frequency domain information and the neural network features. The second to fourth columns are frequency domain information under different thresholds $\mathcal{T}$, where $\mathcal{T}$ is the threshold for removing low frequency. The last three columns are neural network features at different channels. The used model here is VGG16 [37] pre-trained on the ImageNet [38], and the extracted layer is block4_conv3.

III. Method

A. Overview

This section presents the implementation methods and technical details of our proposed framework. Initially, we present a model-agnostic approach for value calculation of unlabeled data, which utilizes the frequency domain to generate multiple candidates. Our method is highly efficient, as frequency domain information can be obtained without model inference, enabling us to deal with large unlabeled datasets and DNNs quickly. Subsequently, we introduce the De-similar Module, which eliminates similar data from each candidate, thereby mitigating the data bias issue. Finally, we introduce the Fast Evaluation Module, which facilitates the selection of informative samples from multiple candidates for AL model training. Compared to current hysteretic AL methods, our Fast Evaluation Module selects the samples that make the greatest contribution to future model learning. Our proposed deep Active Learning framework is shown in Fig. 2.

B. Model-Agnostic Value Calculation

Active learning is a methodology that seeks to identify the most informative data from an unlabeled pool to train a machine learning model. Current Active Learning methods primarily rely on using the model itself to assess the value information of unlabeled data. For instance, EDAL [28] uses evidence learning to estimate the uncertainty of unlabeled samples, where the class probabilities were placed on a Dirichlet distribution. It, therefore, needs to use the AL model to reason about each unlabeled sample. As a result, this line of methods is computationally demanding, particularly when dealing with deep neural networks or large volumes of unlabeled data.

To address this challenge, this paper proposes a model-agnostic approach for estimating the value of unlabeled data. Specifically, we extract frequency domain information from the unlabeled data to quantify its potential value. We observe that the frequency domain information corresponds to object edges, a feature that is known to be important for neural network models, and this has also been discussed in much literature [21], [39]. Hence, we argue that rich frequency domain information is crucial for neural network training. To support this argument, we present a visualization of the frequency domain information and neural network features in Fig. 3. Our approach offers a promising alternative to existing Active Learning methods, and we expect it to be particularly valuable for deep neural network applications and scenarios with large amounts of unlabeled data.

Here we give the details of the proposed model-agnostic value calculation method. Assume $I_{(h,w)}$ is an unlabeled image, where h and w are its height and width. We first use Fourier Transform (FT) to calculate its frequency domain information $F(u,v)$ and remove the low frequency, and the results denote $F(u,v)'$. Then, We use inverse Fourier Transform (iFT) to restore $F(u,v)'$ to image $I_{(H,W)}$. The above process can be calculated as follows:

$$F(u,v)=\sum_{x=0}^{h-1}\sum_{y=0}^{w-1} I_{(h,w)}^{(x,y)} e^{-2j\Pi\left(\frac{ux}{h}+\frac{uy}{w}\right)}, \tag{1}$$

$$F(u,v)' = \begin{cases} F(u,v), & F(u,v) \geq f \\ 0, & F(u,v) < f, \end{cases} \tag{2}$$

$$I_{(H,W)} = \sum_{U=0}^{h-1} \sum_{V=0}^{w-1} F(U,V) e^{2j\Pi\left(\frac{UH}{h} + \frac{VW}{w}\right)}, \tag{3}$$

where $I_{(h,w)}^{(x,y)}$ is the frequency values of coordinate (x, y) in $I_{(h,w)}$. f is the threshold for removing low frequency. Finally, we calculate the sum of the frequency domain values of $I_{(H,W)}$ and denote it as $I_{(H,W)}^{f}$. We regard $I_{(H,W)}^{f}$ as the value of the data $I_{(h,w)}$. We argue that the larger the $I_{(H,W)}^{f}$, the greater the contribution of $I_{(h,w)}$ to model learning. Using the above method, we can calculate the value of all the unlabeled data in the pool and select those with large values for AL model training.

C. De-Similar Module for the Selected Data

Let $Q_{(x,y)}$ and $P_{(x,y)}$ denote the distribution of the unlabeled data pool and the selected data obtained by an AL method, and suppose their densities are $q(x, y) = q(y \mid x)q(x)$ and $p(x, y) = p(y \mid x)p(x)$, respectively. We use $\mathcal{H}(h \sim H)$ to represent the optimal sampling for the original distribution H under the condition of a given sampling rate, where h obeys the distribution H. Based on this definition, $\mathcal{H}((x, y) \sim P_{(x,y)})$ can be calculated as:

$$\begin{aligned} &\mathcal{H}((x,y) \sim Q_{(x,y)}) \\ &\quad = -\iint q(y \mid x)q(x) \ln(q(y \mid x)q(x)) d_x d_y, \end{aligned} \tag{4}$$

$$\begin{aligned} &\mathcal{H}((x,y) \sim P_{(x,y)}) \\ &\quad = -\iint q(y \mid x)q(x) \ln(\mathrm{p}(y \mid x)p(x)) d_x d_y. \end{aligned} \tag{5}$$

We then use KL divergence $D_{KL}(Q_{(x,y)} \mid\mid P_{(x,y)})$ to describe the extent to which $P_{(x,y)}$ covers $Q_{(x,y)}$:

$$\begin{aligned} &D_{KL}(Q_{(x,y)} \mid\mid P_{(x,y)}) \\ &\quad = \mathcal{H}((x,y) \sim P_{(x,y)}) - \mathcal{H}((x,y) \sim Q_{(x,y)}) \\ &\quad = \iint q(y \mid x)q(x) \ln \frac{q(y \mid x)q(x)}{p(y \mid x)p(x)} d_x d_y. \end{aligned} \tag{6}$$

Therefore, we can obtain the optimal Active Learning query function $\mathcal{Q}_{AL}$ by minimizing $D_{KL}(Q_{(x,y)} \mid\mid P_{(x,y)})$:

$$\mathcal{Q}_{AL} = \arg\min_{P_{(x,y)}} D_{KL}(Q_{(x,y)} \mid\mid P_{(x,y)}). \tag{7}$$

However, $P_{(x,y)}$ is often biased towards partial categories in practice. We show a toy example of the data bias problem in Active Learning in Fig. 4, where $P_{(x,y)}$is clearly biased towards the category of the *airplane*. Assuming that the optimal sampling of training data under given conditions $Q_{\widetilde{AL}}$. Obviously, in $P_{(x,y)}$, some high uncertainty data $Q_{\widetilde{AL}} \backslash \mathcal{H}((x,y) \sim P_{(x,y)})$ are not queried by $\mathcal{Q}_{AL}$, but some low uncertainty data $\mathcal{H}((x,y) \sim Q_{(x,y)}) \backslash Q_{\widetilde{AL}}$ are selected instead.

In this paper, we propose the De-similar Module as a solution to the data bias problem. The De-similar Module calculates the degree of similarity between selected data and deletes parts with a high degree of similarity. In particular,

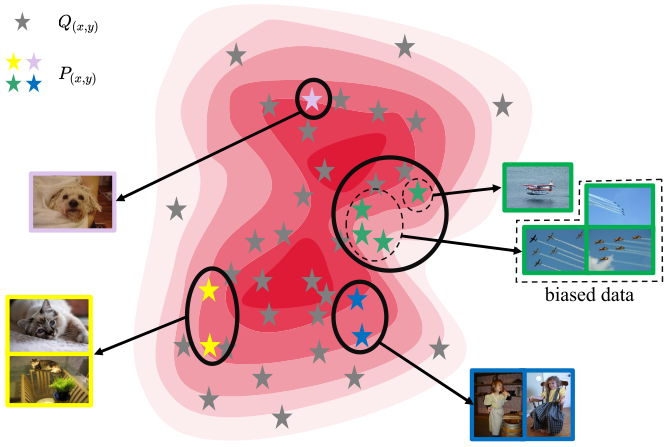


Fig. 4. Data bias problem in Active Learning. The data bias problem refers to the sampling results being biased towards partially similar samples.

for selected data $X \in \mathbb{R}^{(n \times d)}$, we first calculate the degree of similarity between X_i and all other data $X \backslash X_i$, and we denote it as $S(X_i \mid X \backslash X_i)$. Where $S(\cdot \mid \cdot)$ is a similarity calculation kernel. Then, for the similar pair (we use the threshold t to set similar or dissimilar), we remove the one with the low information value from this pair, where the information value is calculated by our proposed model-agnostic value calculation method. The above steps are repeated until there are no data with high similarity to the selected data. In our proposed De-similar Module, we take the Euclidean Distance of the frequency domain as the similarity of the data. The details of the De-similar Module are shown in Algorithm 1, where X and n' are the candidates and the number of required samples in the current AL round, respectively. We use the threshold t to set similarity or dissimilarity. Obviously, threshold t depends on X and n'. Thus, different candidates have different t, and the threshold t in different rounds will also be different. In this sense, our AL framework is dynamic since its selection strategy changes according to the results of model learning. In contrast, other AL methods are static as they select samples following the same strategy at different rounds.

D. Fast Evaluation Module

Active Learning typically involves selecting batches of data from the unlabeled pool for subsequent model training round by round. Usually, unlabeled data are selected based on the training model of the previous round. The quit condition of AL is that the label budget is exhausted or the expected performance of the model is reached. Our approach is inspired by a golden rule in AL: The contribution of the unlabeled data in the i-th round to the model is fixed due to its information being fixed too. Without loss of generality, we use Broad Learning System (BLS) [22] to prove the rule above.

Let $x \in X \in \mathbb{R}^{(n \times d)}$ be the selected data of the i-th round, where n and d are the quantity and dimension of data, respectively. Following the denotations in [23], we consider a continuous function $f \in C\left(\mathbf{I}^d\right)$, which defined on the standard hypercube $\mathbf{I}^d = [0; 1]^d \subset \mathbb{R}^d$. The BLS consists of the feature node outputs f_{w_z} and the enhancement node outputs f_{w_h}, and

Algorithm 1 De-Similar Module

Input: $X \in \mathbb{R}^{(n\times d)}$, n'
Output: $X \in \mathbb{R}^{(n'\times d)}$

Computer the information Inf_X of $X \in \mathbb{R}^{(n\times d)}$, $Inf_X \in \mathbb{R}^{(n\times 1)}$
Calculate the similarity of data in $X \in \mathbb{R}^{(n\times d)}$, similarity $S \in \mathbb{R}^{(n\times n)}$
for i = 1 to n **do**
  for j = 1 to n **do**
    $S\left(X[i], X[j]\right) = \sqrt{\left(\sum X[i] - \sum X[j]\right)^2}$
  end for
end for
Sort the S from large to small, denote the result as $S' \in \mathbb{R}^{n^2\times 1}$,
Get the threshold t, $t = S'\left[n + n'\right]$
for i = 1 to n **do**
  for j = 1 to n **do**
    if $S\left(X[i], X[j]\right) >$ t
      if $Inf_X[i] > Inf_X[j]$
        remove $X[j]$ from $X \in \mathbb{R}^{(n\times d)}$
      if $Inf_X[j] > Inf_X[i]$
        remove $X[i]$ from $X \in \mathbb{R}^{(n\times d)}$
  end for
end for

it can be denoted as:

$$\begin{aligned} f_{w_{m,n}}(\boldsymbol{x}) &= f_{w_z} + f_{w_h} \\ &= \sum_{i=1}^{n\times k} w_i \phi\left(\boldsymbol{x}\boldsymbol{w}_{e_i} + \beta_{e_i}\right) \\ &\quad + \sum_{j=1}^{m\times q} w_{nk+j}\xi\left(\boldsymbol{z}\boldsymbol{w}_{h_j} + \beta_{h_j}\right), \end{aligned} \tag{8}$$

where ϕ and ξ are nonconstant bounded feature mapping and activation function, respectively. $n \times k$ and $m \times q$ are the numbers of feature nodes f_{w_z} and enhancement nodes f_{w_h}, and w_z and w_h are their weights to the output. $\boldsymbol{w}_{e_i}$, $\boldsymbol{w}_{h_j}$, and bias β are randomly generated. Assume that those random variables are defined on the probability measure $\mu_{m,n}$, and notation E is the expectation with respect to the probability measure. BLS can improve the learning ability of the model by adding feature nodes and enhancement nodes. As such, the improved BLS $f_{w_{m,n}}(\boldsymbol{x})^+$ can be denoted as:

$$\begin{aligned} f_{w_{m,n}}(\boldsymbol{x})^+ &= (f_{w_z} + f_{w_z}^+) + (f_{w_h} + f_{w_h}^+) \\ &= \sum_{i=1}^{n\times k+1} w_i \phi\left(\boldsymbol{x}\boldsymbol{w}_{e_i} + \beta_{e_i}\right) \\ &\quad + w_{n\times k+1}\phi\left(\boldsymbol{x}\boldsymbol{w}_{e_{n\times k+1}} + \beta_{e_{n\times k+1}}\right) \\ &\quad + \sum_{j=1}^{m\times q+1} w_{nk+j}\xi\left(\boldsymbol{z}\boldsymbol{w}_{h_j} + \beta_{h_j}\right) \\ &\quad + w_{nk+m\times q+1}\xi\left(\boldsymbol{z}\boldsymbol{w}_{h_{m\times q+1}} + \beta_{h_{m\times q+1}}\right), \end{aligned} \tag{9}$$

where $f_{w_z}^+$ and $f_{w_h}^+$ are the incremented feature node outputs and the incremented enhancement node outputs. Thus, the distance between $f_{w_{m,n}}$ and $f_{w_{m,n}}^+$ on the compact set $\mathbf{K} \subset \mathbf{I}^d$ can be denoted as:

$$\begin{aligned} &\rho_{\mathbf{K}}\left(f_{w_{m,n}}, f_{w_{m,n}}^+\right) \\ &\quad = \sqrt{E\left[\int_{\mathbf{K}} \left(f_{w_{m,n}}(x) - f_{\boldsymbol{w}_{m,n}}(x)^+\right)^2 dx\right]}. \end{aligned} \tag{10}$$

To solve this formula, we first denote $f_{d_{w_z}} = f - f_{w_z}$. According to the theories in [40], there exists a function $f_{c_{w_z}} \in C\left(\mathbf{I}^d\right)$, and:

$$\rho_{\mathbf{K}}\left(f_{d_{w_z}}, f_{c_{w_z}}\right) < \frac{\varepsilon}{4}, \tag{11}$$

where $\forall\varepsilon > 0$. In some cases, we need to increase the number of feature nodes to improve the capabilities of the BLS, and we denote the added feature nodes as $f_{w_z}^+$. According to the universal approximation property in [35], there exists a sequence of $f_{w_z}^+$, and:

$$\rho_{\mathbf{K}}\left(f_{c_{w_z}}, f_{w_z}^+\right) < \frac{\varepsilon}{4}. \tag{12}$$

For the case of adding enhancement nodes, we can get similar conclusions: $\rho_{\mathbf{K}}\left(f_{d_{w_h}}, f_{c_{w_h}}\right) < \frac{\varepsilon}{4}$ and $\rho_{\mathbf{K}}\left(f_{c_{w_h}}, f_{w_h}^+\right) < \frac{\varepsilon}{4}$. Where $f_{d_{w_h}} = f - f_{w_z} - f_{w_z}^+$, and $f_{c_{w_h}}$ is a function similar to $f_{c_{w_z}}$.

Based on the above analysis, we can finally get:

$$\begin{aligned} &\rho_{\mathbf{K}}\left(f_{w_{m,n}}, f_{w_{m,n}}^+\right) \\ &= \sqrt{E\left[\int_{\mathbf{K}} \left(f_{w_{m,n}}(\boldsymbol{x}) - f_{\boldsymbol{w}_{m,n}}(\boldsymbol{x})^+\right)^2 d\boldsymbol{x}\right]} \\ &= \sqrt{E\left[\int_{\mathbf{K}} \left(f_{w_z}(x) + f_{w_h}(x) - f_{w_z}^+(x) - f_{w_h}^+(x)\right)^2 d\boldsymbol{x}\right]} \\ &\le \rho_{\mathbf{K}}\left(f_{w_z}, f_{w_z}^+\right) + \rho_{\mathbf{K}}\left(f_{w_h}, f_{w_h}^+\right) \\ &\le \rho_{\mathbf{K}}\left(f_{d_{w_z}}, f_{c_{w_z}}\right) + \rho_{\mathbf{K}}\left(f_{c_{w_z}}, f_{w_z}^+\right) \\ &\quad + \rho_{\mathbf{K}}\left(f_{d_{w_h}}, f_{c_{w_h}}\right) + \rho_{\mathbf{K}}\left(f_{c_{w_h}}, f_{w_h}^+\right) \\ &\le \frac{\varepsilon}{4} + \frac{\varepsilon}{4} + \frac{\varepsilon}{4} + \frac{\varepsilon}{4} \\ &= \varepsilon. \end{aligned} \tag{13}$$

Therefore, for the data $X \in \mathbb{R}^{(n\times d)}$ selected in the i-th round, there exists $f_{w_{m,n}}$ such that $\rho_{\mathbf{K}}\left(f_{w_{m,n}}, f_{w_{m,n}}^+\right) = \varepsilon$. As a result, we need to select a batch of unlabeled data that contains as much information as possible. We naturally seek the AL method to achieve the above goal. However, AL methods are currently hysteretic, in which samples are selected based on the model learned previously. Although those methods can benefit from the training achievements in the previous model, the learned model often judges the value of unlabeled data in a lag way (e.g., uncertainty estimation based on the previous model) and thus can not ensure that the selected data can contribute to future learning. Our Fast Evaluation Module solves this problem by evaluating the feature contribution of the selected candidate directly and leveraging the BLS to speed up the evaluation process.

Here we introduce in detail how the Fast Evaluation Module calculates the contribution of candidates to future model learning. Let $X_{train} \in \mathbb{R}^{(n_1 \times d)}$ denote the training data selected and labeled in the previous rounds, and $X_{remain} \in \mathbb{R}^{(n_2 \times d)}$ denote the remaining unlabeled data. For candidates $\mathcal{X} = [X_1, X_2, \cdots, X_n]$, $X_i \in \mathbb{R}^{(n \times d)}$, we first use the model to obtain its pseudo-label and denote it as $\mathcal{X}' = [X'_1, X'_2, \cdots, X'_n]$, $X'_i \in \mathbb{R}^{(n \times (d+1))}$. We define a model kernel $\odot(X, Y)$ to represent the result of the model trained on X and predicated on Y. Therefore, for candidate $X_i \in \mathbb{R}^{(n \times d)}$, we can roughly estimate its contribution Δ_{X_i} to future model learning, and it can be calculated as:

$$\Delta_{X_i} = \odot(X_{\text{train}} + X'_i, X_{\text{remain}}) - \odot(X_{\text{train}}, X_{\text{remain}}). \quad (14)$$

Finally, the contribution of candidates is calculated, and the largest one is selected as the data for the current round.

IV. Experiment

This section first shows the experiment preliminaries, i.e., dataset, models, parameter settings, and evaluation metrics. Then, we introduce the training process of the proposed AL framework and analyze the experimental results of the classification task. Finally, we did the ablation study to prove the effectiveness of several key designs in FastAL, including the model-agnostic value calculation module, the De-similar Module, and the Fast Evaluation Module.

A. Experiment Preliminaries

1) Datasets: **CIFAR-10 [41].** CIFAR-10 contains 60k images (50k for training and 10k for testing), grouped into 10 categories (*airplane*, *automobile*, *bird*, *cat*, *deer*, *dog*, *frog*, *horse*, *ship*, *truck*), each with 6k images.

CIFAR-100 [41]. CIFAR-100 contains 60k images (50k for training and 10k for testing) with 20 super-categories (*aquatic mammals*, *fish*, *flowers*, *food containers*, *fruit and vegetables*, *household electrical devices*, etc.), each super-category contains 5 categories (e.g., the categories of the *super-category fish* are *aquarium fish*, *flatfish*, *ray*, *shark*, *trout*), and each category has 600 images.

MNIST [36]. MNIST is a handwritten digit dataset with 10 categories (the digits *0*∼*9*), and it contains 70k images (60k for training and 10k for testing).

Fashion-MNIST [42]. Fashion-MNIST is a fashion products dataset with 10 categories, and it contains 70k images (60k for training and 10k for testing). Fashion-MNIST is more difficult than the original MNIST dataset.

2) The Used Models and Parameter Settings: The used models include InceptionV3 [43], ResNet-50 [44], and MobileNetV3 [45], and their corresponding implementations rely on the pre-existing open-source repository[1] of TensorFlow. Moreover, the BLS[2] in Fast Evaluation Module also adheres to the official implementation guidelines. We did not make any changes to the above baseline network structures in pursuit of a fair comparison. All experiments were trained with the same settings: optimizer is SGD, weight decay = 0.00004, decay factor of learning rate is 0.94, learning rate = 0.01, momentum = 0.9, batch-size = 32.

3) Evaluation Metrics: Active learning aims to minimize the cost of data labeling and has two typical applications in practical scenarios. The first application scene is to select the most informative batch of unlabeled data within a fixed labeling budget to optimize the model's performance. The second application scene is to use minimal labeled data while maintaining the expected model performance. We evaluate the proposed Active Learning framework based on these two realistic scenarios using two metrics, with the first being the prevalent metric in the Active Learning literature [11], [12], [16], [17], [20], [29], [30] and the second being a novel metric introduced in this paper.

Metric 1: Performance under a fixed labeling budget. A higher model performance indicates a better Active Learning framework when evaluated using this metric. The labeling budget we set during the experiment does not exceed 50% of the original unlabeled data since Active Learning is usually employed when labeling costs are constrained.

Metric 2: Labeling budget under expected model performance. The sampling approaches that minimize labeling costs are preferred when evaluated using this metric. We set different expected performances for different models to ensure low labeling costs.

Most existing Active Learning methods primarily rely on Metric 1 to assess the effectiveness of proposed frameworks. However, given the practical relevance of Metric 2, a simultaneous evaluation of both metrics can facilitate a more comprehensive comparison of the strengths and limitations of various Active Learning methods.

B. Evaluation on Classification

1) Comparing Baselines: We compare our proposed Active Learning framework FastAL with the following baseline:

a) Random sample: Samples are randomly selected from the unlabeled pool.

b) The uncertainty-based AL methods: Sparse Modeling Active Learning (SMAL) [11] is an AL method that can alleviate the data bias problem because it considers the diversity and density of data at the same time. It is worth noting that SMAL partitions the dataset into multiple subsets, including the seed set (labeled set), the unlabeled set, the validation set, and the testing set. In order to ensure fairness of comparison, we use random sampling in the first round, and the settings of other rounds follow the original paper. Ensembles-based Active Learning (ENS) [12] uses an ensemble network to calculate data uncertainty. To avoid unbalanced initialization, ENS uses label information when selecting the samples in the first round, and this is obviously unreasonable. Also, for the fairness of comparison, ENS in our experiment and FastAL use the same samples in the first AL round.

c) The diversity-based AL methods: Fisher Kernel Self Supervision (FKSS) [16] uses the proposed feature density matching methods to predict the contribution of unlabeled data to future training. For the four datasets in the experiment, we use all the data instead of using only part of the data, like

[1] https://github.com/tensorflow/models/tree/master/research/slim
[2] https://broadlearning.ai

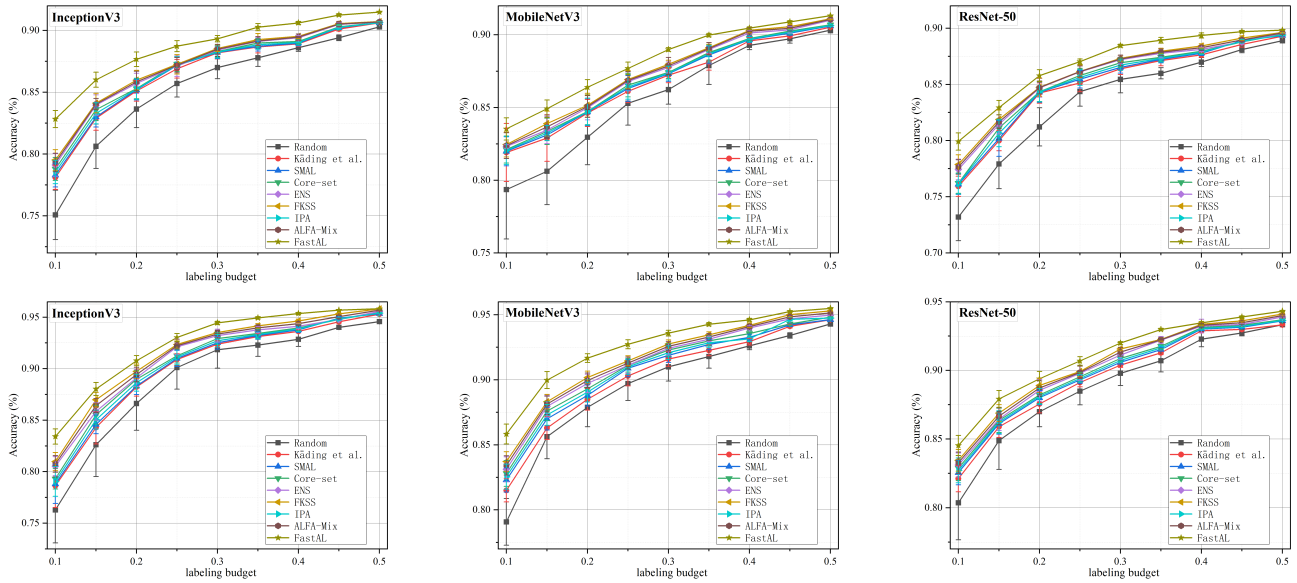


Fig. 5. Performance on CIFAR-10 [41] (top) and MNIST [36] (bottom) under fixed labeling budget. The evaluation networks used here are InceptionV3 [43], ResNet-50 [44], and MobileNetV3 [45]. The compared recent state-of-the-art AL frameworks included uncertainty-based methods ENS [12] and SMAL [11], diversity-based methods Core-set [17], FKSS [16], IPA [29], and ALFA-Mix [30], expected model change AL methods Käding et al. [20], and random sample. We repeat the experiment five times and report the mean and the standard deviation (error bar).

FKSS, to artificially create an unlabeled dataset that obeys a specific distribution. For the other diversity-based AL methods, Core-set [17], IPA [29], and ALFA-Mix [30], we follow the training tricks and hyperparameters in the original paper.

d) The expected model change AL methods: Käding et al. [20] use the approximated gradients of the loss function to enable the Gaussian process regression AL methods to be used for deep neural network models. Following the original paper, we also use a stochastic gradient approximation with just a single sample to estimate model parameter updates, and the models we use are all DNNs to ensure that the baseline can keep its advantages, thereby ensuring fair comparison.

2) Active Learning Settings: We conducted experiments on two metrics, i.e., performance under fixed labeling budget and labeling budget under expected model performance. For metric 1, we evaluated FastAL and other baseline methods on the CIFAR-10 and MNIST. We first select 5% of the unlabeled data (About 3000 for CIFAR-10 and 3500 for MNIST) as the training data for all AL methods in the first round. Then, we use FastAL and other baseline methods to select the same amount of data (5% of the unlabeled pool) for the other rounds. We repeat the above steps until the selected data reaches the labeling budget. In this paper, we set the labeling budget is 50%. For metric 2, we evaluated FastAL and other baseline methods on CIFAR-100 and Fashion-MNIST. Also, we first select 5% of the unlabeled data (About 3000 for CIFAR-100 and 3500 for Fashion-MNIST) as the training data for all methods in the first round. Then, we use FastAL and other baseline methods to select the same amount of data for the other rounds. We repeat the above steps until the learned model reaches the expected model performance. Considering that different models are suitable for different scenarios, we set different expected model performances for these models, i.e., for CIFAR-100, expected model performances are 0.7, 0.75, and 0.8, for Fashion-MNIST, expected model performances are 0.85, 0.9, and 0.95.

The following section shows how to generate different Fast Evaluation Module candidates. Take the selection of 5% data from the unlabeled pool as an example. We first calculate the value of unlabeled data through the proposed model-agnostic value calculation and take out the top 10%, top 15%, top 20%, top 25%, and top 30% data as candidates. Then we use the De-similar Module to remove part of the similar data from the above candidates until only the same amount of data remains for each candidate (5% of the unlabeled pool). Thus, for different candidates, the threshold t in Algorithm 1 is also different. Generally, the larger the amount of data in the candidate, the smaller the t. Finally, We directly use the Fast Evaluation Module to calculate the training results of candidates to select the samples that will contribute the most to the future model learning from these candidates.

3) Results and Analysis: Based on the experimental settings in Section IV-A, we compared FastAL and various types of baselines. We evaluated our AL framework on both metric 1 (performance under fixed labeling budget) and metric 2 (labeling budget under expected model performance) and reported the results in Fig. 5 and Table I, respectively. Furthermore, FastAL incorporates BLS as an evaluation module instead of utilizing the baseline AL model. This design was primarily based on the exceptional efficiency of BLS. To substantiate this claim, we present the inference time per image in Table II, as well as the total overhead per AL round, accounting for varying latencies across different methodologies. From these results, we have several observations:

(1) Our proposed FastAL framework outperforms all baselines by a clear margin. To be specific, under metric 1, models based on the FastAL framework have a higher performance under all fixed labeling budgets (see Fig. 5). Also, under

TABLE I

LABELING BUDGET ON CIFAR-100 [41] AND FASHION-MNIST [42] UNDER DIFFERENT EXPECTED MODEL PERFORMANCES. THE EVALUATION NETWORKS USED HERE ARE INCEPTIONV3 [43], RESNET-50 [44], AND MOBILENETV3 [45]. THE COMPARED RECENT STATE-OF-THE-ART AL FRAMEWORKS INCLUDED UNCERTAINTY-BASED METHODS ENS [12] AND SMAL [11], DIVERSITY-BASED METHODS CORE-SET [17], FKSS [16], IPA [29], AND ALFA-MIX [30], EXPECTED MODEL CHANGE AL METHODS KÄDING ET AL. [20], AND RANDOM SAMPLE. WE REPEAT THE EXPERIMENT FIVE TIMES AND REPORT THE MEAN. THE FIRST AND SECOND BEST RESULTS ARE COLORED IN RED AND BLUE, RESPECTIVELY

	InceptionV3 [43]						ResNet-50 [44]						MobileNetV3 [45]					
	CIFAR-100			Fashion-MNIST			CIFAR-100			Fashion-MNIST			CIFAR-100			Fashion-MNIST		
	0.70	0.75	0.80	0.85	0.9	0.95	0.70	0.75	0.80	0.85	0.9	0.95	0.70	0.75	0.80	0.85	0.9	0.95
Random	5.2	6.8	7.8	3.4	4.4	5.2	5.2	6.8	7.8	4.2	5.2	5.8	5.2	6.8	7.8	3.6	4.6	5.2
Käding et al. [20]	4.8	5.8	6.8	2.8	3.6	4.4	5.0	6.0	7.0	3.6	4.4	5.4	4.8	5.8	7.0	2.8	3.6	4.6
SMAL [11]	4.6	5.6	6.8	2.8	3.4	4.4	4.8	5.8	6.8	3.6	4.4	5.4	4.6	5.6	6.8	2.8	3.6	4.4
Core-set [17]	4.6	5.6	6.4	2.6	3.4	4.4	4.8	5.6	6.6	3.4	4.4	5.4	4.6	5.6	6.6	2.8	3.6	4.4
ENS [12]	4.4	5.4	6.4	2.6	3.4	4.2	4.6	5.6	6.4	3.4	4.4	5.2	4.6	5.4	6.4	2.8	3.6	4.2
FKSS [16]	4.2	5.2	6.2	2.6	3.4	4.2	4.4	5.4	6.4	3.2	4.2	5.2	4.4	5.4	6.2	2.6	3.4	4.2
IPA [29]	4.6	5.6	6.6	2.8	3.4	4.4	4.8	5.8	6.8	3.4	4.4	5.4	4.6	5.6	6.6	2.8	3.6	4.4
ALFA-Mix [30]	4.4	5.4	6.2	2.6	3.4	4.2	4.6	5.4	6.4	3.4	4.2	5.2	4.6	5.4	6.4	2.6	3.4	4.2
FastAL	4.0	5.0	6.0	2.0	3.0	4.0	4.0	5.0	6.2	3.0	4.0	5.0	4.0	5.0	6.0	2.0	3.0	4.0

TABLE II

INFERENCE TIME (MS) PER IMAGE AND PER AL ROUND

	InceptionV3	ResNet-50	MobileNetV3	BLS
Per image	3.2893	2.8815	1.2544	0.0034
Per AL round	12828.28	9768.286	4628.736	11.4256

TABLE III

THE PERFORMANCE UNDER DIFFERENT COMPONENT COMBINATIONS. C.1, C.2, AND C.3 STAND FOR THE FAST EVALUATION MODULE, THE MODEL-AGNOSTIC VALUE CALCULATION MODULE, AND THE DE-SIMILAR MODULE. WITHOUT THE COMP.1 MEANS THAT WE USE CANDIDATE 1 DIRECTLY IN EACH ROUND. THE ABSENCE OF THE COMP.2 MODULE MEANS THAT WE GENERATE MULTIPLE CANDIDATES THROUGH RANDOM SAMPLING IN EACH ROUND. WITHOUT THE COMP.3 MEANS THAT WE WILL NOT REMOVE DATA WITH HIGH SIMILARITY AMONG THE SELECTED DATA

	C.1	C.2	C.3	CIFAR-10	MNIST
InceptionV3				0.9150	0.9583
ResNet-50	✓	✓	✓	0.9132	0.9550
MobileNetV3				0.8983	0.9432
InceptionV3				0.9129	0.9568
ResNet-50		✓	✓	0.9111	0.9531
MobileNetV3				0.8962	0.9412
InceptionV3				0.9036	0.9469
ResNet-50	✓		✓	0.9041	0.9453
MobileNetV3				0.8812	0.9289
InceptionV3				0.9012	0.9453
ResNet-50	✓	✓		0.8812	0.9316
MobileNetV3				0.8856	0.9307

metric 2, models based on the FastAL framework require less labeled data to reach the expected performance (see Table I).

(2) Models based on the FastAL framework will have a more stable training process. We repeat the experiment five times and report the mean, and standard deviation (see error bar in Fig. 5) as the training data for all methods in the first round is randomly selected. Note that our value calculation method is model-agnostic. Although we can use it to select valuable data in the first round to get higher performance, we still adopt random sampling in the first round for the fairness of comparison. From the standard deviation in Fig. 5, we can see that the stability of our method is much better than other baselines. We argue this is primarily due to the De-similar Module that we propose, which reduces data bias and improves the performance and stability of model training. We will further prove this in the ablation study.

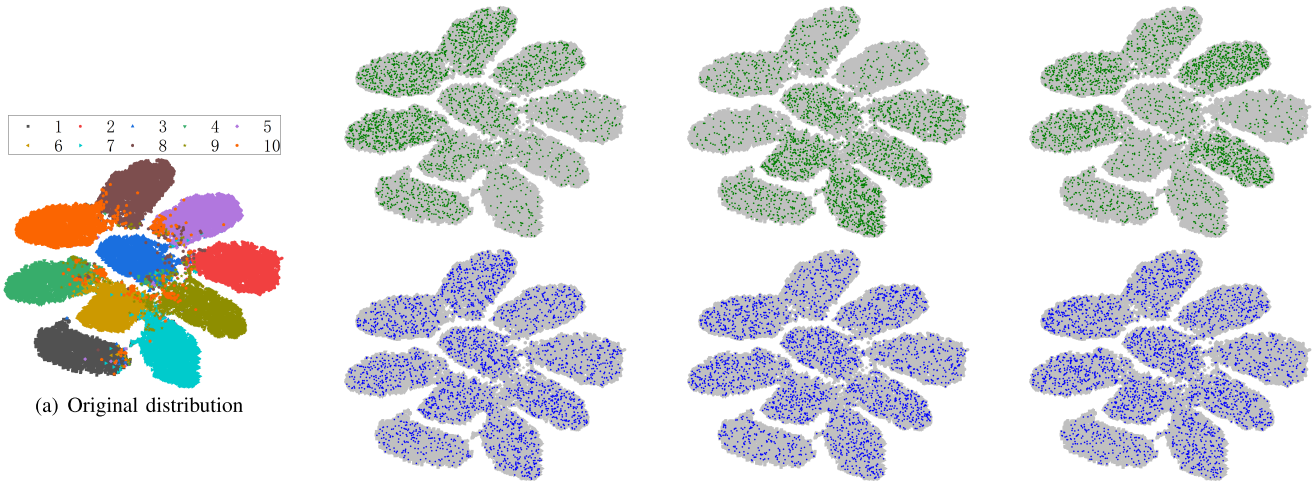
(3) Our proposed FastAL offers more obvious advantages in the early rounds of AL. The early rounds are often more in line with the needs of AL in actual application scenarios due to its labeling budget and expected model performance is lower. Thus, our proposed method will have more advantages in actual application scenarios.

(4) In some cases, there will a round gap between FastAL and other baseline methods, i.e., FastAL uses fewer rounds to achieve the same expected performance (e.g., for CIFAR-100 [41], MobileNetV3 [45] with FastAL framework uses 6.0 rounds while Käding et al. [20] need 7.0 to meet the expected performance of 0.8). Due to the round gap, our framework can use less labeled data and fewer training rounds to achieve the training goal, thus saving both labeling and training costs.

(5) Compared to the direct utilization of the baseline AL model for computing data contribution to future training, the BLS we have introduced significantly mitigates time overhead. According to Table II, BLS can reduce overhead by 367-966 times for each image. Moreover, in view of the distinct latencies associated with diverse approaches, we have also presented the aggregate overhead for each AL round, which demonstrates that BLS can economize on the cost by 404-1121 times.

C. Ablation Study

As described in section III, our work is mainly composed of three components, including the De-similar Module, the model-agnostic value calculation module, and the Fast Evaluation Module. This section conducts ablation studies to confirm the effectiveness of these components. We reported the experiment results of different component combinations



(a) Original distribution (b) Round 1 (c) Round 2 (d) Round 3

Fig. 6. T-SNE [46] embedding of selected samples in different AL rounds (b-d) and full dataset (a), the former feature is extracted from the output of the model learned by the AL models, and the latter is from Broad Learning System (BLS) [22]. The top (green points) and bottom (blue points) are the sampling results of the baseline method (Sparse Modeling Active Learning, SMAL [11]) and our proposed AL framework, respectively.

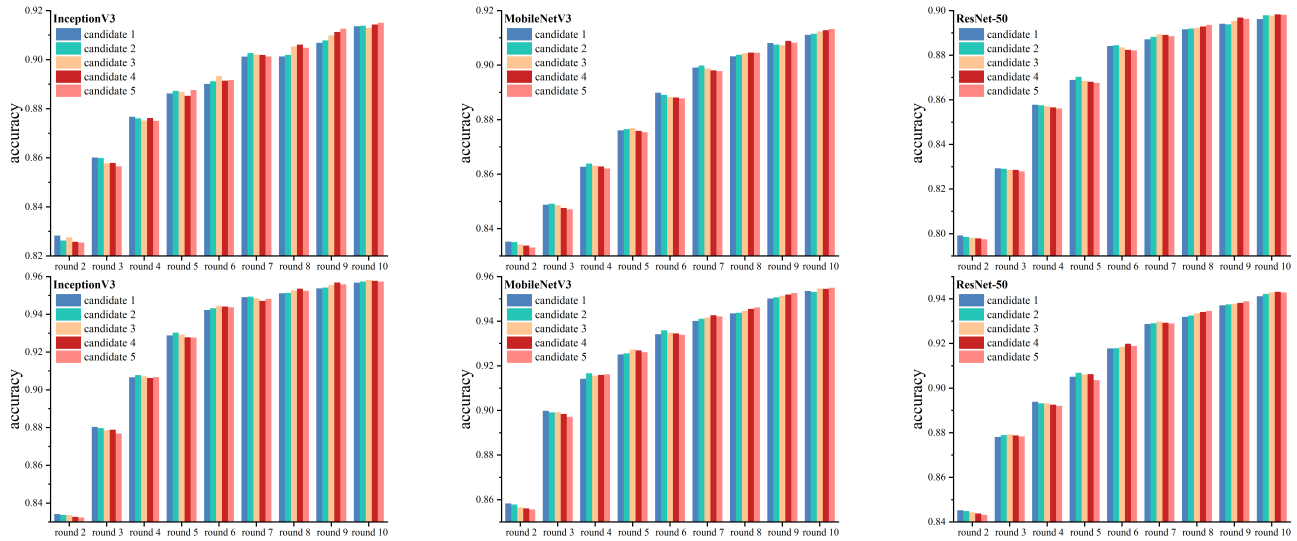


Fig. 7. The performances of different candidates in different rounds. The rules for generating different candidates are the same as those in Section IV-B, i.e., calculate the value of unlabeled data through the proposed model-agnostic value calculation, and take out the top 10%, top 15%, top 20%, top 25%, and top 30% data as candidates.

in Table III, and we can conclude from the results that each component contributes to the FastAL. In the following, we will explore the benefits of these components further.

1) Without the De-Similar Module: The De-similar module in our proposed framework can remove some high-similarity samples, thus alleviating the data bias problem. To demonstrate this benefit, we visualize the t-SNE [46] embedding of selected samples in different AL rounds and the full dataset in Fig. 6. Fig. 6 shows that the samples selected by the baseline method [11] suffer from the data bias problem (green points in Fig. 6). For example, the sampling results of the baseline method in round 1 are biased towards categories *3*, *7*, and *10* while ignoring categories *1*, *6*, and *8*. Furthermore, we can observe that the data bias problem will continue to affect the model learning in subsequent rounds. For instance, in round 2, the baseline method ignored the biased categories (*3*, *7*, and *10*). We argue this is mainly because the learned model will focus on learning the partial categories because of the data bias in round 1; these categories, therefore, will be ignored in the subsequent rounds. However, the visualization results show that our FastAL framework can significantly alleviate the data bias problem, i.e., the samples it selects are not significantly biased toward any partial category (blue points in Fig. 6). We believe that this debiased sampling can effectively prevent the model from converging to partial categories and thus improve the performance of the model.

2) Without the Fast Evaluation Module: The data selection of current AL methods often relies on a single standard (e.g., data uncertainty in the uncertainty-based methods), and the standard will not change during this learning process. However, it may not always be the selected data based on a single standard that contributes the most to future training.

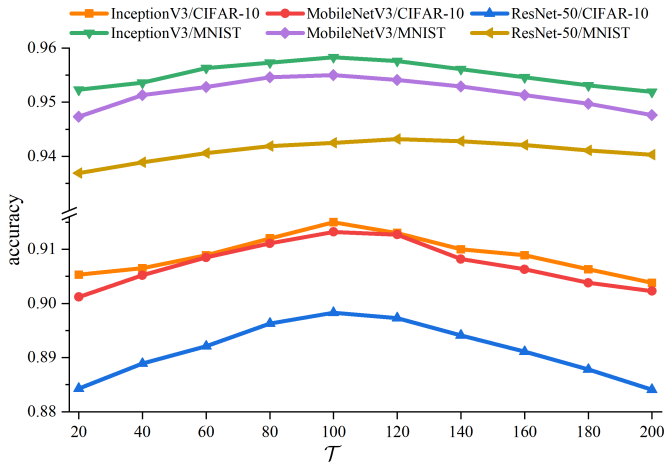


Fig. 8. The performances under different thresholds $\mathcal{T}$, where $\mathcal{T}$ is the threshold for removing low frequency. The smaller the $\mathcal{T}$, the more the frequency domain information is retained, but the more noise is retained at the same time. The smaller the $\mathcal{T}$, the less frequency domain information is retained, but at the same time, the less noise is retained.

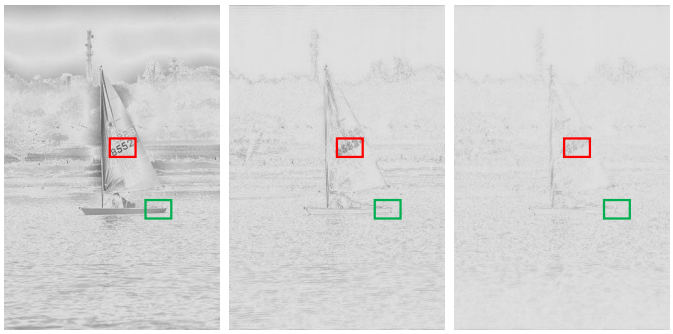


Fig. 9. The frequency domain information under different thresholds $\mathcal{T}$. The red boxes and the green boxes represent noise and useful information, respectively.

Although the data information calculation method we proposed is also based on a single standard (based on the frequency domain information), we generate multiple candidates through different data selection mechanisms and directly select the one that contributes the most to future training through the Fast Evaluation Module. Thus, the standards in the FastAL framework will adaptively change as the model learning progresses, as the final candidates selected at different rounds may be different. We report the performance of different candidates at different rounds in Fig. 7, and it shows that the optimal candidate changes dynamically at different rounds as expected. Our proposed Fast Evaluation Module aims to select the best candidate, so it can naturally contribute to the FastAL.

3) Threshold $\mathcal{T}$ in the Model-Agnostic Value Calculation Module: In this paper, we propose the model-agnostic value calculation module, which calculates the value of unlabeled data from the perspective of the frequency domain without relying on model reasoning. As a result, in the face of large-scale unlabeled data and DNNs, our method has a huge cost advantage over model-dependent methods. From Table III, we can see that the model-agnostic value calculation module is indeed effective. $\mathcal{T}$ is the threshold for frequency domain information processing in our model-agnostic value calculation method. We report the model performance under different thresholds $\mathcal{T}$ in Fig. 8 to further explore its influence. We can see that the performance will decrease when $\mathcal{T}$ is very large or very small. In Fig. 9, we also showed the frequency domain information of the data under different thresholds $\mathcal{T}$ to find the cause of the above phenomenon. From the presented samples, we argue that when the $\mathcal{T}$ is large, the key information will be lost (see green boxes in Fig. 9), and when the $\mathcal{T}$ is small, a lot of useless noise will remain (see purple boxes in Fig. 9). We set $\mathcal{T}$ to 100 in this paper.

V. Conclusion

We present FastAL, a novel and efficient dynamic deep Active Learning framework comprising three key components: the Fast Evaluation Module, the model-agnostic value calculation module, and the De-similar Module. Our proposed framework demonstrates superior performance compared to various state-of-the-art AL methods, including uncertainty-based, diversity-based, and expected model change AL methods. Specifically, our model-agnostic value calculation method enables cost-effective processing of large-scale unlabeled data and Deep Neural Networks (DNNs) without relying on model inference to calculate the value of the data. Our Fast Evaluation Module directly calculates the contribution of data to future rounds of training, which is distinct from existing methods that primarily rely on the learned model from the current round. Furthermore, the De-similar Module alleviates the data bias problem by removing partial data with high similarity. Our ablation study confirms the effectiveness of these components.

We demonstrate the effectiveness of our framework on the classification task and believe that it has the potential to be applied to more complex visual tasks, such as pedestrian re-identification, object detection, and segmentation. These will be the focus of our future work.

References

[1] B. Brattoli, J. Tighe, F. Zhdanov, P. Perona, and K. Chalupka, "Rethinking zero-shot video classification: End-to-end training for realistic applications," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4612–4622.

[2] N. Han et al., "Double relaxed regression for image classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 2, pp. 307–319, Feb. 2020.

[3] X. Chen, H. Li, Q. Wu, K. N. Ngan, and L. Xu, "High-quality R-CNN object detection using multi-path detection calibration network," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 2, pp. 715–727, Feb. 2021.

[4] Y. Xu et al., "Gliding vertex on the horizontal bounding box for multi-oriented object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 4, pp. 1452–1459, Apr. 2021.

[5] L. Yang, H. Li, F. Meng, Q. Wu, and K. N. Ngan, "Task-specific loss for robust instance segmentation with noisy class labels," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 1, pp. 213–227, Jan. 2023.

[6] Z. Tan et al., "Real time video object segmentation in compressed domain," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 1, pp. 175–188, Jan. 2021.

[7] C. Oh and A. Cavallaro, "View-action representation learning for active first-person vision," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 2, pp. 480–491, Feb. 2021.

[8] M. Rong, H. Cui, Z. Hu, H. Jiang, H. Liu, and S. Shen, "Active learning based 3D semantic labeling from images and videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 12, pp. 8101–8115, Dec. 2022.

[9] H. Xu, S. Zhi, S. Sun, V. M. Patel, and L. Liu, "Deep learning for cross-domain few-shot visual recognition: A survey," 2023, *arXiv:2303.08557*.

[10] X. Jiang et al., "DFEW: A large-scale database for recognizing dynamic facial expressions in the wild," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 2881–2889.

[11] G. Wang, J. Hwang, C. Rose, and F. Wallace, "Uncertainty-based active learning via sparse modeling for image classification," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 316–329, Jan. 2019.

[12] W. H. Beluch, T. Genewein, A. Nurnberger, and J. M. Kohler, "The power of ensembles for active learning in image classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9368–9377.

[13] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *Proc. Conf. Res. Develop. Inf. Retr.*, 1994, pp. 3–12.

[14] E. Elhamifar, G. Sapiro, A. Yang, and S. S. Sasrty, "A convex optimization framework for active learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 209–216.

[15] A. J. Joshi, F. Porikli, and N. Papanikolopoulos, "Multi-class active learning for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 2372–2379.

[16] D. Gudovskiy, A. Hodgkinson, T. Yamaguchi, and S. Tsukizawa, "Deep active learning for biased datasets via Fisher kernel self-supervision," in *Proc. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 9038–9046.

[17] O. Sener and S. Savarese, "Active learning for convolutional neural networks: A core-set approach,," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–13.

[18] B. Settles, M. Craven, and S. Ray, "Multiple-instance active learning," in *Proc. Neural Inf. Process. Syst.*, 2008, pp. 1289–1296.

[19] A. Freytag, E. Rodner, and J. Denzler, "Selecting influential examples: Active learning with expected model output changes," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2014, pp. 562–577.

[20] C. Käding, E. Rodner, A. Freytag, and J. Denzler, "Active and continuous exploration with deep neural networks and expected model output changes," 2016, *arXiv:1612.06129*.

[21] C. C. Kao, T. Y. Lee, P. Sen, and M. Y. Liu, "Localization-aware active learning for object detection," in *Proc. Asian Conf. Comput. Vis.*, 2019, pp. 506–522.

[22] C. L. P. Chen and Z. Liu, "Broad learning system: An effective and efficient incremental learning system without the need for deep architecture," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 10–24, Jan. 2018.

[23] C. L. P. Chen, Z. Liu, and S. Feng, "Universal approximation capability of broad learning system and its structural variations," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 1191–1204, Apr. 2019.

[24] D. Yoo and I. S. Kweon, "Learning loss for active learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 93–102.

[25] K. Xu, M. Qin, F. Sun, Y. Wang, Y. Chen, and F. Ren, "Learning in the frequency domain," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1737–1746.

[26] Y. Zhong, B. Li, L. Tang, S. Kuang, S. Wu, and S. Ding, "Detecting camouflaged object in frequency domain," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 4494–4503.

[27] W. Huang et al., "Unsupervised fusion feature matching for data bias in uncertainty active learning," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Oct. 10, 2022, doi: 10.1109/TNNLS.2022.3209085.

[28] S. Sun, S. Zhi, J. Heikkilä, and L. Liu, "Evidential uncertainty and diversity guided active learning for scene graph generation," in *Proc. Int. Conf. Learn. Represent.*, 2023, pp. 1–22.

[29] R. Mahmood, S. Fidler, and M. T. Law, "Low budget active learning via Wasserstein distance: An integer programming approach," in *Proc. Int. Conf. Learn. Represent.*, 2022, pp. 1–31.

[30] A. Parvaneh, E. Abbasnejad, D. Teney, R. Haffari, A. Van Den Hengel, and J. Q. Shi, "Active learning by feature mixing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 12227–12236.

[31] S. Xu, J. Liu, C. Yang, X. Wu, and T. Xu, "A learning-based stable servo control strategy using broad learning system applied for microrobotic control," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 13727–13737, Dec. 2022.

[32] B. Sheng, P. Li, R. Ali, and C. L. P. Chen, "Improving video temporal consistency via broad learning system," *IEEE Trans. Cybern.*, vol. 52, no. 7, pp. 6662–6675, Jul. 2022.

[33] X. Lin, S. Sun, W. Huang, B. Sheng, P. Li, and D. D. Feng, "EAPT: Efficient attention pyramid transformer for image processing," *IEEE Trans. Multimedia*, vol. 25, pp. 50–61, 2023.

[34] X. Jiang, Y. Zong, W. Zheng, J. Liu, and M. Wei, "Seeking salient facial regions for cross-database micro-expression recognition," in *Proc. 26th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2022, pp. 1019–1025.

[35] B. Igelnik and Y.-H. Pao, "Stochastic choice of basis functions in adaptive function approximation and the functional-link net," *IEEE Trans. Neural Netw.*, vol. 6, no. 6, pp. 1320–1329, Nov. 1995.

[36] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[37] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[38] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 248–255.

[39] J. Xu, C. Shi, C. Qi, C. Wang, and B. Xiao, "Unsupervised part-based weighting aggregation of deep convolutional features for image retrieval," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 7436–7443.

[40] W. Rudin, *Real and Complex Analysis (Higher Mathematics Series)*. New York, NY, USA: McGraw-Hill, 2015.

[41] A. Krizhevsky et al., "Learning multiple layers of features from tiny images," Toronto, ON, Canada, 2009.

[42] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.

[43] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[45] A. Howard et al., "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1314–1324.

[46] G. Hinton and S. Roweis, "Stochastic neighbor embedding," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2002, pp. 857–864.

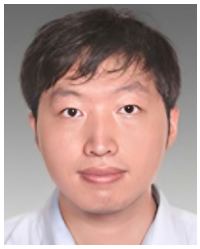
Shuzhou Sun is currently pursuing the Ph.D. degree with the Center for Machine Vision and Signal Analysis (CMVS), Faculty of Information Technology and Electrical Engineering, University of Oulu, Oulu, Finland. His current research interests include computer vision, deep learning, and image processing.

Huali Xu is currently pursuing the Ph.D. degree with the Center for Machine Vision and Signal Analysis (CMVS), Faculty of Information Technology and Electrical Engineering, University of Oulu, Oulu, Finland. Her current research interests include computer vision, deep learning, and few-shot learning.

Yan Li received the Ph.D. degree from the Software Engineering Institute, East China Normal University, China, in 2022. He is currently a Lecturer with the Department of Computer Science, Shanghai Normal University (SHNU), Shanghai, China. His current research interests include designing new algorithms for geometric deep learning, using techniques from information geometry, stochastic analysis, and optimal control, for solving problems, involving learning and reasoning in high-dimensional non-Euclidean structured data, such as graphs and manifolds.

Ping Li (Member, IEEE) received the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong, Hong Kong, in 2013.

He is currently an Assistant Professor with the Department of Computing and an Assistant Professor with the School of Design, The Hong Kong Polytechnic University, Hong Kong. He has published many scholarly research articles, pioneered several new research directions, and made a series of landmark contributions in his areas. He has an excellent research project reported by the ACM TechNews, which only reports the top breakthrough news in computer science worldwide. More importantly, however, many of his research outcomes have strong impacts to research fields, addressing societal needs and contributed tremendously to the people concerned. His current research interests include image/video stylization, colorization, artistic rendering and synthesis, realism in non-photorealistic rendering, computational art, and creative media.



Bin Sheng (Member, IEEE) received the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong, Shatin, Hong Kong, in 2011.

He is currently a Full Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. His current research interests include virtual reality and computer graphics. He is an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.

Xiao Lin received the Ph.D. degree in computer science from Shanghai Jiao Tong University, Shanghai, China.

She is currently a Full Professor with the College of Information, Mechanical and Electrical Engineering, Shanghai Normal University (SHNU), Shanghai. She is also a Visiting Scholar with the Department of Computer Science and Engineering, Shanghai Jiao Tong University. Her current research interests include image processing, computer vision, and machine learning.