FAA-CLIP: Federated Adversarial Adaptation of CLIP

Yihang Wu, Ahmad Chaddad, Christian Desrosiers, Tareef Daqqaq, Reem Kateb

Abstract—Despite the remarkable performance of vision language models (VLMs) such as Contrastive Language Image Pre-training (CLIP), the large size of these models is a considerable obstacle to their use in federated learning (FL) systems where the parameters of local client models need to be transferred to a global server for aggregation. Another challenge in FL is the heterogeneity of data from different clients, which affects the generalization performance of the solution. In addition, natural pre-trained VLMs exhibit poor generalization ability in the medical datasets, suggests there exists a domain gap. To solve these issues, we introduce a novel method for the Federated Adversarial Adaptation (FAA) of CLIP. Our method, named FAA-CLIP, handles the large communication costs of CLIP using a light-weight feature adaptation module (FAM) for aggregation, effectively adapting this VLM to each client's data while greatly reducing the number of parameters to transfer. By keeping CLIP frozen and only updating the FAM parameters, our method is also computationally efficient. Unlike existing approaches, our FAA-CLIP method directly addresses the problem of domain shifts across clients via a domain adaptation (DA) module. This module employs a domain classifier to predict if a given sample is from the local client or the global server, allowing the model to learn domain-invariant representations. Extensive experiments on six different datasets containing both natural and medical images demonstrate that FAA-CLIP can generalize well on both natural and medical datasets compared to recent FL approaches. Our codes are available at https://github.com/AIPMLab/FAA-CLIP.

Index Terms—Federated learning, Foundation models, Domain adaptation.

I. Introduction

While models based on deep learning (DL) have achieved ground-breaking results in a broad range of computer vision and natural language understanding tasks, their performance is often dependent on the availability of large datasets [1]. In recent years, there has been a growing concern on ensuring data privacy and security, with many organizations implementing regulations and laws such as the EU General Data Protection Regulation (GDPR) [2]. These restrictions on sharing raw data from different organizations poses a siginificant challenge for training robust DL models in fields like medical imaging where privacy is of utmost importance. One of the most promising solutions to this problem is federated learning (FL).

Y. Wu and A. Chaddad are with the School of Artificial Intelligence, Guilin University of Electronic Technology, Guilin, China.
Corresponding author: Ahmad Chaddad.
T. Daqqaq is with the College of Medicine, Taibah University, Al Madinah, Saudi Arabia.
R. Kateb is with the College of Computer Science and Engineering, Jeddah University, Jeddah, Saudi Arabia.
C. Desrosiers and A. Chaddad are with The Laboratory for Imagery, Vision and Artificial Intelligence, Ecole de Technologie Superieure, Montreal, Canada. Email: ahmad8chaddad@gmail.com, ahmadchaddad@guet.edu.cn

FL is a decentralized training framework that aggregates many local models into a global model without sharing raw data, thereby complying with the requirements for data security [3], [4].

The standard approach for FL consists in training a local model at each client involved in the process, using the client's own data, and then broadcasting the parameters of local models to a global server. The server then combines these parameters to obtain a global model, the parameters of which are sent back to the clients. Repetition of this process for several rounds allows the global model to implicitly learn from the data of different clients, without having to directly share these data [5]. Although FL models improve the privacy and security of DL systems trained from multiple sources, their performance and efficiency may be affected by two factors: **1)** the communication and computation resources needed for global transmission and local training; **2)** data distribution shifts among the clients. For example, discrepancies in the distribution of data (i.e., data heterogeneity) can influence the performance of FL models, especially in medical imaging where the data acquisition process and patient demographics can vary largely from one site (e.g., hospital or clinic) to another [6], [7]. This challenge can limit the generalizability of FL models where the test data are from different sources with large heterogeneity [8]. Recent studies have shown that VLMs trained on natural images do not perform well on classification tasks involving medical images, highlighting the need for specialized adaptation mechanisms [9]. Moreover, the communication costs associated with parameter transmission can be prohibitive if the model is large, which poses a challenge for applications that require timely communication where high transmission latency is not allowed [10]. This is especially problematic for VLMs such as CLIP that have more than 10^8 parameters (version based on ViT-B/32) [11].

This study investigates the potential of natural pretrained VLMs in the context of FL and propose a novel framework called Federated Adversarial Adaptation of CLIP (FAA-CLIP) addressing the challenges of data heterogeneity, communication, and computation costs. FAA-CLIP handles the large size of CLIP with a light-weight feature adaptation module (FAM) that adapts this VLM to each client's data. Keeping the parameters of CLIP frozen, our method only updates and broadcasts the parameters of the clients' FAM, thereby reducing the overall computational and communication costs. Furthermore, compared to existing FL approaches, FAA-CLIP addresses the problem of data heterogenity in different clients directly, using a specialized domain adaptation (DA) module. The proposed module enables FAA-CLIP to learn domain-

invariant representations with a domain classifier that predicts whether a given sample is from the local client or from a global reference data set. The novelty of this study lies in our proposal of a training and communication efficient FL approach using a lightweight FAM and a domain adaptation module for adapting CLIP to both natural and medical datasets. Further calibration analysis conducted demonstrates the potential of FAA-CLIP for calibration tasks.

The contributions of our work can be summarized as follows:

1) We propose a specialized feature adaptation module (FAM) for adapting CLIP to natural and medical imaging data from multiple sites, which significantly reduces the computational and communication costs of our solution.
2) Unlike existing apporoaches like FedCLIP [11], our FAA-CLIP method addresses the problem of domain shifts directly using a domain adaptation (DA) module. As shown in our results, this DA module boosts performance in the various test scenarios.
3) Through experiments on six publicly available classification benchmark datasets, including both natural and medical images, we demonstrate the higher classification accuracy and balanced accuracy of FAA-CLIP over state-of-the-art FL approaches.

II. RELATED WORK

Before presenting our FAA-CLIP method, we give a brief overview of related work on federated learning and CLIP-based methods in the literature.

Federated learning (FL). The increasing awareness of challenges related to data privacy and security has urged researchers to develop novel machine learning solutions that can learn from multiple sources of data without having to share this data [2]. Among the many solutions proposed for this issue, FL has emerged as the most promising and popular approach [12]. Methods based on this approach can be roughly divided into three categories: horizontal federated learning (HFL), vertical federated learning (VFL), and federated transfer learning (FTL) [6]. In HFL, clients share the same feature space while holding different samples. On the other hand, VFL in the FL setting where the datasets of individual clients contain information about the same samples but have different features. Last, in the FTL setting, client datasets differ both in terms of features and samples, with limited overlaps.

One of the leading HFL methods, FedAVG [5], proposes a simple averaging mechanism to combine the parameters of locally-trained models into a single global one. Although it preserves data privacy by only sharing parameters, this method may still suffer from the problem of data heterogeneity between clients [13]. To solve this issue, FedProx added a proximal regularization term and allowed slight model gaps between clients and the server [14]. In [15], authors argued that local distribution shifts in separate clients can be handled efficiently via batch normalization (BN) parameters. Based on this idea, they proposed a FedBN method that updates the BN parameters of each client locally without averaging them

on the global server. On the other hand, MOON [16] used a contrastive loss between local clients and global servers to improve overall performance.

Despite their improved performance, these methods mostly ignored the problem of generalization, which is also an important challenge in FL. To solve this problem, the work in [17] proposed a method to address separately the performance gaps arising from unseen client data (out-of-sample gap) and from unseen client distributions (participation gap). Likewise, the authors of [18] introduced a FL framework based on the Sharpness Aware Minimization (SAM) local optimizer to improve the model's generalization ability. Moreover, the method presented in [19] employed a new variance reduction regularizer to dynamically calibrating the aggregation weights, thereby improving generalization ability. In [20], they proposed a novel FL framework using mutual learning to provide a personalized local model with feasible performance. However, it introduces extra local training epochs. Although these approaches boost the model robustness to distribution shifts, they were not intended for foundation models such as the one used in our work.

Contrastive Language Image Pre-training (CLIP) is an effective and scalable method learning representations jointly from image and text data [21]. The CLIP model, which was trained from scratch using over 400 million image-text data pairs collected from the Internet [21], recently demonstrated remarkable performance in a wide range of image understanding tasks [22]. However, the integration and use of VLMs like CLIP in FL applications is stll in its infancy. In [23], authors proposed replacing full VLM training with federated text prompt learning to reduce communication costs. Nevertheless, significant computational costs are still required to train the VLM. The work in [24] also highlighted the challenge of designing task-specific prompts for CLIP. Furthermore, in [11], an adapter-based strategy was designed to leverage the power of pre-trained VLMs. This strategy, which only tunes and aggregates the parameters of adapters, demonstrated its effectiveness on several natural datasets. Inspired by the superior performance of CLIP for zero shot and few shot tasks, the authors of [25] used CLIP to optimize FL between the server and client models, alleviating the problem of data heterogeneity between clients. However, this approach incurred large communication costs as it requires sharing the entire model. In [26], they introduced low-rank adaptation (LoRA) for large language models to reduce computational costs, and this technique is widely used in the CLIP backbone in recent studies [27]. In [28], they proposed to use personalized prompts instead of global prompts for aggregation. The main idea is that they decoupled the parameters into base (global) and personalized (local) parameters. In [29], they proposed an adapter-based dual teacher technique using mutual knowledge distillation to adapt VLMs in FL tasks. However, it has not been validated in medical datasets. Furthermore, in [23], they proposed to aggregate the prompts instead of a shared global model to reduce communication costs (PromptFL).

Unlike previous studies, we explore the potential of natural pre-trained CLIP in the FL context for both natural and medi-

cal images with an acceptable resource cost. We also introduce calibration analysis [30] into CLIP-based FL approaches, an area that has been less explored in this context.

III. METHODOLOGY

Before presenting our FAA-CLIP method for the federated adversarial adaptation of CLIP, we first introduce the FL setting considered in our study.

A. Problem formulation

In our FL setting, we work with a set of N clients $\{C_1, C_2, \cdots, C_N\}$, each client C_i having a private dataset $\mathcal{D}_i = \{(\mathbf{x}_{i,j}, y_{i,j})\}_{j=1}^{n_i}$. As in related studies [11], we assume that the data of separate clients have the same input and output space, but follow different distributions, i.e., $P(\mathcal{D}_{i'}) \neq P(\mathcal{D}_i)$, $\forall i' \neq i$. Each dataset $\mathcal{D}_i$ consists of three non-overlapping parts, namely a training $\mathcal{D}_i^{train} = \{(\mathbf{x}_{i,j}^{train}, y_{i,j}^{train})\}_{j=1}^{n_i^{train}}$, a validation $\mathcal{D}_i^{val} = \{(\mathbf{x}_{i,j}^{val}, y_{i,j}^{val})\}_{j=1}^{n_i^{val}}$ and a test set $\mathcal{D}_i^{test} = \{(\mathbf{x}_{i,j}^{test}, y_{i,j}^{test})\}_{j=1}^{n_i^{test}}$. Our goal is to learn a global model $f_\theta(\cdot)$ while preserving data privacy and security. This model provides a good performance on the test data of every client, i.e.,

$$\min_f \ \frac{1}{N} \sum_{i=1}^{N} \frac{1}{n_i^{test}} \sum_{j=1}^{n_i^{test}} \ell(f_\theta(\mathbf{x}_{i,j}^{test}), y_{i,j}^{test}), \quad (1)$$

based on a given loss function ℓ. For generalization, we assume that there exist Q different clients $\{M_1, M_2, \cdots, M_Q\}$ with data $\mathcal{D}_i^M = \{(\mathbf{x}_{i,j}, y_{i,j})\}_{j=1}^{n_i}$. Our objective is for the model to perform on clients that *were not included* in the local training phase, i.e.,

$$\min_f \ \frac{1}{M} \sum_{i=1}^{M} \frac{1}{m_i} \sum_{j=1}^{m_i} \ell(f_\theta(\mathbf{x}_{i,j}), y_{i,j}). \quad (2)$$

B. Our FAA-CLIP framework

Our FL framework for CLIP-based image classification, which is illustrated in Figure 1, comprises three key components: 1) a *feature adaptation module* (FAM) for efficient updating and sharing of models, 2) a *feature adaptation strategy* addressing the problem of data distribution shifts between clients, and 3) a *global aggregation strategy* to combine the learned features from multiple clients. We present each of these components in the following.

Feature Adaptation Module (FAM) training. We use a pretrained CLIP model, comprising an image encoder $e_I(\cdot)$ and a text encoder $e_T(\cdot)$, to extract features from the data for each client C_i. For a training example $\mathbf{x}_j \in \mathcal{D}_i^{train}$, we denote as $\mathbf{I}_j = e_I(\mathbf{x}_j) \in \mathbb{R}^D$ the D-dimensional vector of image features. For text features, we use the standard prompt "a picture of a {class}" as input to the text encoder to obtain features $\mathbf{T}_j = e_T(\mathbf{x}_j) \in \mathbb{R}^D$.

Pre-trained VLMs can extract a rich set of features, however, not all of those are suitable for learning a specific task. This is particularly true for detecting and classifying abnormal regions such as lesions in medical images, as these regions are absent in normal images and typically represent a small part of the image. To identify the regions of focus for locally trained models, we introduce a local FAM, denoted as $att_i(\cdot)$. The main idea of FAM is to reduce the training costs while adapting the features extracted by CLIP to the new task through a series of nonlinear transformations. This FAM takes as input image features $\mathbf{I}$ and returns an attention mask $att_i(\mathbf{I}) \in [0,1]^D$. The attention mask is typically generated using the Softmax function. The mask is then used to generate the features of the masked images $\mathbf{I}^* = att_i(\mathbf{I}) \otimes \mathbf{I}$, where $\otimes$ is the Hadamard product (element-wise).

We measure the probability that an example $\mathbf{x}_j$ belongs to a class c using the cosine similarity between the image features of $\mathbf{x}_j$ and the text features $\mathbf{T}_c$ corresponding to the prompt of c:

$$p(\mathrm{Y} = c \,|\, \mathbf{x}_j) \,=\, \frac{\exp(s_{j,c}/\tau)}{\sum_{c'=1}^{K} \exp(s_{j,c'}/\tau)}, \ \text{ with } s_{j,c} = \frac{\langle \mathbf{I}_j^*, \mathbf{T}_c \rangle}{\|\mathbf{I}_j^*\| \cdot \|\mathbf{T}_c\|} \quad (3)$$

where τ is the Softmax temperature parameter.

Keeping the image and text encoders frozen, we train the local FAMs by minimizing a contrastive loss $\mathcal{L}_{contr}$ that pushes together the image and text features from the same training example and pulls apart non-matching ones. Following [21], we compute the contrastive loss over batches of size B. Let $\mathbf{S}$ be the $B \times B$ matrix where $s_{j,j'}$ is the cosine similarity between the image features $\mathbf{I}_j^*$ and $\mathbf{T}_{j'}$ as measured in Eq (3). We compute an image probability matrix $\mathbf{P} = softmax(\mathbf{S}/\tau) \in [0,1]^{B \times B}$ and a text probability matrix $\mathbf{Q} = softmax(\mathbf{S}^\top/\tau) \in [0,1]^{B \times B}$. The contrastive loss is then formulated as follows:

$$\mathcal{L}_{contr} \,=\, -\frac{1}{B} \sum_{j=1}^{B} \frac{1}{2} \Big(\log p_{j,j} + \log q_{j,j} \Big). \quad (4)$$

Feature adaptation. We use a domain adversarial loss to align feature representations between clients [31]. For each local client i, this loss trains a domain classifier (*discriminator*) D_i that predicts whether a given image representation $\mathbf{I}_j$ is from a *source* ($D_i(\mathbf{I}_j) = 1$) or a *target* domain ($D_i(\mathbf{I}_j) = 0$). For client i, the source domain contains the image features $\mathbf{I}_j$ of the client's training samples, i.e. $\mathbf{x}_j \in \mathcal{D}_i$. Since private data cannot be shared in our FL scenario, we use a global set of unlabeled images $\mathcal{D}_{tgt}$ to obtain the representations of the target domain. This requirement does not constitue a significant limitation as there are many publicly available datasets in medical imaging (e.g., UK Biobank [32] or Human Connectome Project – HCP [33] for brain images). For each client i, batches of $2B$ samples are generated by randomly selecting the same number of source and target examples. Denoting as $z_j = \mathbb{1}(\mathbf{x}_j \in \mathcal{D}_i)$ the domain label of example $\mathbf{x}_j$, the domain adaptation (DA) loss is then defined using cross-entropy, as follow:

$$\mathcal{L}_{DA} \,=\, -\frac{1}{2B} \sum_{j=1}^{2B} z_j \log D_i(\mathbf{I}_j^*) + (1 - z_j) \log \big(1 - D_i(\mathbf{I}_j^*)\big) \quad (5)$$

This loss is minimized for the domain classifier D_i, but maximized for the adapter att_i so that the adapted image
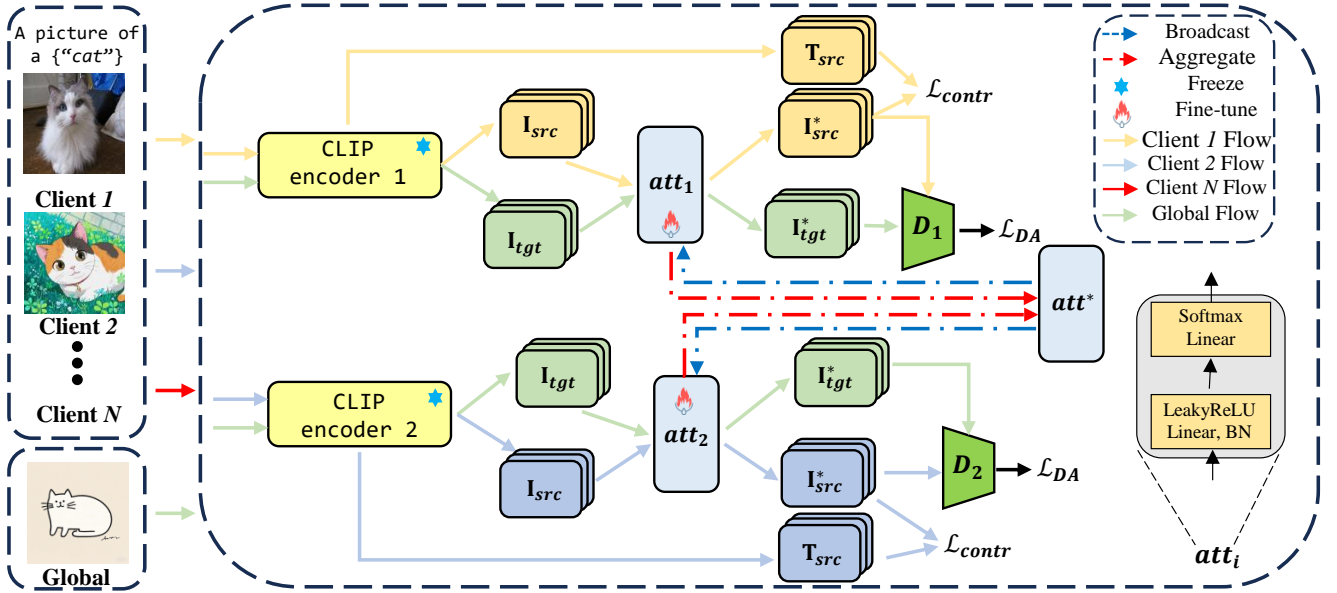
Fig. 1: FAA-CLIP pipeline. Each client trains its local model separately, optimizing only the parameters of its local feature adaptation module (FAM) (att_i) and domain classifier D_i using contrastive and domain adaptation losses. After receiving the local client parameters, the server aggregates them into a global (FAM) (att^*) whose parameters are transmitted back to clients.

features of each client's data are indistinguishable from those on the shared reference data. Combined with the contrastive loss of Eq. (3), this gives rise to the following adversarial objective:

$$\min_{att_i} \max_{D_i} \mathcal{L} = \mathcal{L}_{contr}(att_i) - \lambda \mathcal{L}_{DA}(att_i, D_i), \quad (6)$$

where λ is a hyper-parameter controlling the trade-off between these two loss terms. In practice, this minimax optimization problem is converted to standard minimization using a gradient reversal layer [31].

Algorithm 1 Training procedure of FAA-CLIP.

Require: N clients' data $\{\mathcal{D}_i\}_{i=1}^N$, image encoder e_I, text encoder e_T, feature adaptation module att^*, domain classifier D_i
Ensure: Feature adaptation module att
1: For client i, compute the corresponding features obtain image and text features $I = e_I(image)$, $T = e_T(text)$
2: For client i, train the local feature adaptation module, att_i and domain classifier D_i according to Eq. (5);
3: Send the current feature adaptation module att_i to the server;
4: Aggregate feature adaptation modules' parameters via Eq. (7) to obtain att^*;
5: Transmit att^* to each client i;
6: **Repeat** steps 2 to 5 until convergence;

Global aggregation. The last component of our FAA-CLIP framework is the aggregation strategy to combine the parameters of different clients into a single global model. This strategy works as follows. In every round, each client C_i uploads its FAM parameters θ_i^{att} to the server. Thereafter, the server combines these parameters into a single vector θ_{global}^{att} using a simple average

$$\theta_{global}^{att} = \frac{1}{N} \sum_{i=1}^N \theta_i^{att}. \quad (7)$$

Subsequently, the server redistributes the global FAM parameters to every client. Compared to CLIP encoders, the FAM has less parameters. Hence our method has very low computational and communication costs. Algorithm 1 summarizes the main steps of FAA-CLIP.

IV. EXPERIMENTS

We start by presenting the data used for evaluating the proposed FAA-CLIP model and then provide implementation details about our model's architecture, parameter optimization strategy and FL setup. Afterwards, we demonstrate our method's outstanding performance by comparing it to state-of-the-art FL approaches and baselines. Finally, we present several ablation studies to further investigate the impact of our method's different components and hyperparameters.

A. Datasets

Our method is evaluated on six datasets that contain both natural and medical images, which are described in the following.

OfficeHome is a large image classification benchmark, which contains 65 classes [34]. This dataset has four sub-domains (Art, Clipart, Product, and Real World) with about 15,500 images. Following [11], for each client, we divide the data into training (60%), validation (20%), and test (20%) sets. In each experiment, three subdomains are used as clients (denoted as source), while one subdomain is considered as the global testing set (denoted as target).

Multi-OF is a custom dataset derived from OfficeHome dataset. Following [20], Dirichlet distribution is used to generate client data. Specifically, A, C and P domain are used for local training, while each domain is divided into five clients. For each domain data, we considered different α values used
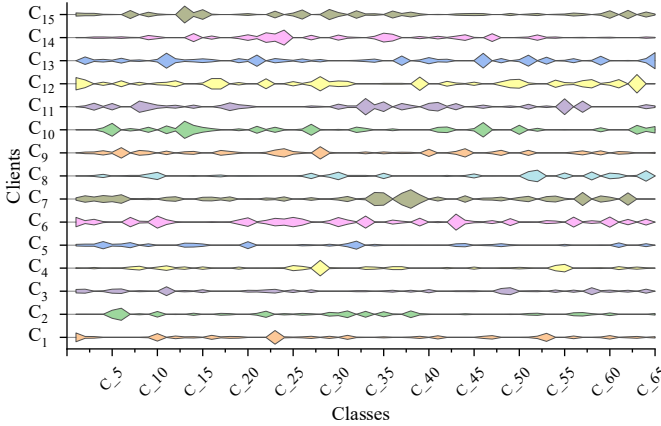
Fig. 2: Example of data distribution in each client using kite graph in Multi-OF dataset. C_1 to C_{15} indicate each client while C_1 to C_65 represent each class.

TABLE I: Samples of each client.

Dataset	Clients			
OfficeHome	A (2,427)	C (4,365)	P (4,439)	R (4,357)
ModernOffice31	A (2,817)	D (498)	S (3,100)	W (795)
SC	C_1 (1,971)	C_2 (19,766)	C_3 (8,512)	C_{glo} (6,233)
BT	C_1 (1,474)	C_2 (423)	C_3 (949)	C_{glo} (394)
HK	C_1 (3,186)	C_2 (3,186)	C_3 (3,186)	C_{glo} (1,104)

in the Dirichlet function to simulate heterogeneous data. Furthermore, domain R is used for the global evaluation following the study in [11]. Figure 2 shows the data distribution in each client using the kite graph (e.g., the size of polygon reflects the number of samples). As illustrated, there exist large heterogeneous data across all clients.

ModernOffice31 is a refined version of the Office31 image classification benchmark [35] having four domains, Amazon (A), Webcam (W), DSLR (D) and Synthetic (S), with 31 classes in each domain. For every client, we divide the data into three parts: a training set (80%), and a testing set (20%). Following OfficeHome, we view each sub-dataset as a client and choose three sub-datasets as sources while the rest serves as the target client. We evaluate our and other methods on ModernOffice31 in a similar way to OfficeHome.

Brain tumor (BT) is a public Kaggle dataset, which has four different classes, namely glioma tumor, meningioma tumor, no tumor, and pituitary tumor[1]. The training set is composed of 2,870 samples, while the test set has 394 samples. We randomly split the training set into three clients, and each client's data is then divided into two subsets: a training set (80%), a validation set (10%), and a testing set (10%). Finally, we evaluated the global model using the original testing set.

Multi-source skin cancer (SC). We build this dataset from three sources: a kaggle public dataset, HAM10000, and ISIC2019 [36]–[38]. Since ISIC2019 lacks a test set, we divided it into two parts, one for training and the other for testing, with a ratio of 8:2. We selected the common classes for this dataset: actinic keratosis (AK), basal cell carcinoma

[1]https://www.kaggle.com/dsv/1183165

(BCC), dermatofibroma (DF), melanoma (MEL), nevus (NV), pigmented benign keratosis (PBK), and vascular lesion (VL). For each client, we divide the data into two parts: a training set (80%) and a testing set (20%).

HyperKvasir (HK). To further evaluate our model, we use the largest gastrointestinal dataset [39], comprising 10,662 labeled images with 23 classes. We randomly divide the dataset into four clients with the same number of samples, and we use three clients for training and employ the rest for testing. For each client in training, we split the data into two parts: a training set (70%) and a testing set (30%).

A detailed description of the number of samples in each client, for all datasets, is provided in Table I.

TABLE II: The computation overhead (all client's local training time per communication round) and the communication overhead (transmitted parameters per round). Σ, Σ_f, Σ_{fc} and Σ_{fam} are the parameter amount in the CLIP backbone, the CLIP without last layer, the fully-connection layer and the FAM, respectively. Σ_{prompt} is the parameter amount in prompt learner.

	Computation (mins)				Communication
	OfficeHome	ModernOffice31	BT	SC	Param./round
FedAVG	0.63	1.01	0.34	2.06	$2 \times \Sigma$
FedProx	0.75	1.12	0.39	2.25	$2 \times \Sigma$
MOON	0.74	1.18	0.41	2.21	$2 \times \Sigma$
LoRA$_{r=3}$	0.60	0.98	0.32	2.03	$2 \times 0.98\% \times \Sigma$
PromptFL	0.49	0.87	0.15	2.05	$2 \times \Sigma_{prompt}$
FedCLIP	0.44	0.79	0.16	2.05	$2 \times \Sigma_{fam}$
Ours	0.52	0.93	0.23	2.06	$2 \times \Sigma_{fam}$

B. Implementation details

We adopted the pre-trained ViT-B/32 model as backbone for the image encoder in our CLIP-based framework. Our light-weight FAM for adaptation is composed of five layers: a first linear layer, a batch normalization layer, a LeakyReLU layer, a second linear layer, and a softmax activation function. The domain classifier D comprises eight layers: a first linear layer, a batch normalization layer, a ReLU layer, a second linear layer, a batch normalization layer, a ReLU layer, a third linear layer and a sigmoid activation function. During training, we keep the CLIP encoders frozen and only optimize the parameters of the FAM and local domain classifier D_i using the Adam optimizer with betas set to 0.9 and 0.98, a weight decay of 0.02, a fixed learning rate of 5×10^{-5}, an epsilon of 1×10^{-6}, and a batch size of 32 [40], [41]. For all datasets, the hyperparameter λ is set to 0.5 in Eq. (6).

To have a comprehensive comparison, we include four recent FL approaches in our experiments: FedAVG [5], FedProx [14], MOON [16], LoRA [42], PromptFL [23] and FedCLIP [11]. The same experimental setting described above is used for all tested methods. In the case of FedProx, the parameter μ_{Prox} is set to $1e^{-2}/2$, while for MOON, we set μ_{MOON} as 1. For LoRA, we set the low rank as 3 ($r = 3$). Moreover, we fixed the random seed in PyTorch as 0 to eliminate the variation caused by different seeds. For the OfficeHome
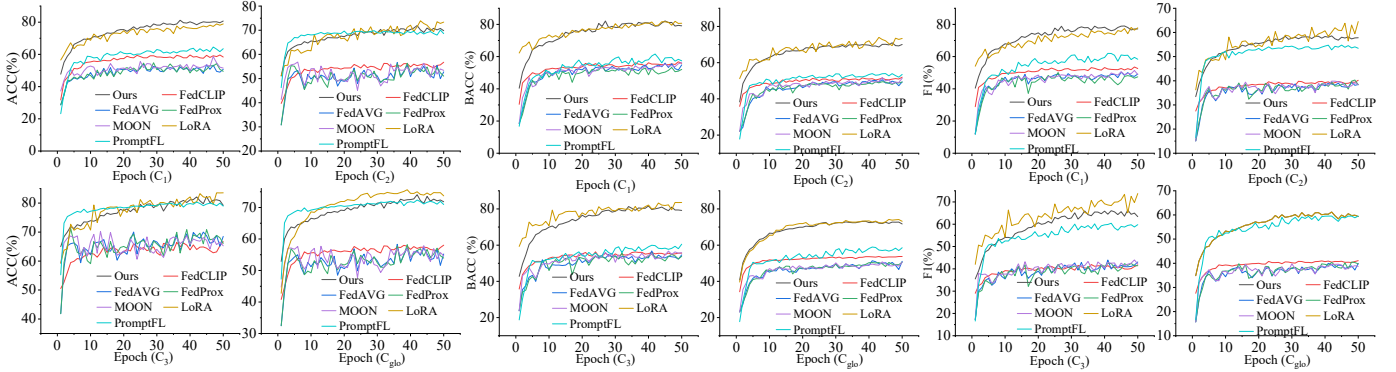
Fig. 3: Testing accuracy (ACC), balanced accuracy (BACC) and macro-F1 for each communication epoch using skin cancer (SC) dataset for FAA-CLIP, FedProx, FedAVG, FedCLIP, LoRA$_{r=3}$, PromptFL and MOON.
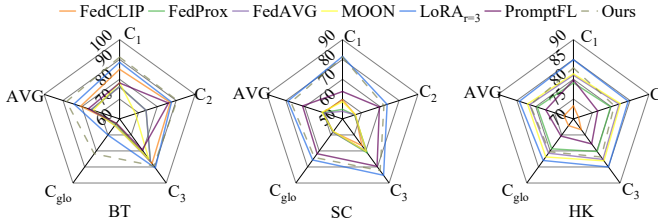

Fig. 4: Testing accuracy of baselines and FAA-CLIP for BT, SC and HK datasets.

dataset, the results of all baselines except MOON are based on [11].

For FL, we set the number of global training rounds to 50 for all datasets. For each round, we perform a single epoch of local training and aggregate the parameters of all clients. For image pre-processing, we resized the images to 224×224, and normalized their intensity using z-score normalization for both the training and testing phases. The testing environment used for the experiments is based on the Windows 11 operating system, and features an Intel 13900KF CPU with 128 GB of RAM and an RTX 4090 GPU. We use PyTorch 1.13.1 with Python 3.8. The scikit-learn library[2] was used to calcuate all metrics. We evaluated performance using the top-1 classification accuracy (ACC) as the primary metric. Furthermore, for the SC, Dermnet and HK datasets, we also consider balanced accuracy (BACC) (average of recall obtained on each class) and macro F1 score (average of F1 obtained on each class) as metrics due to the class imbalance in these datasets.

C. Results

Communication and computational costs. We record the time cost of each round for each method and the communication overhead in each round on OfficeHome ({ACPR}), ModernOffice31 ({ADSW}), BT and SC, as reported in Table II. As illustrated, FAA-CLIP is training efficient compared to traditional FL techniques such as FedAVG, FedProx, and MOON, while it indicates lower computational costs compared to LoRA. Furthermore, the communication costs for our

[2]https://scikit-learn.org/stable/modules/model_evaluation.html#classification-metrics

TABLE III: Accuracy(%) in the OfficeHome dataset. **Bold** means the best.

	Source			Target	Avg
	A	C	P	R	
FedCLIP	78.35	68.38	87.94	87.79	80.61
FedProx	70.93	68.73	77.73	75.42	73.20
FedAVG	70.93	68.73	77.73	75.42	73.20
MOON	67.42	69.30	76.32	75.81	72.21
LoRA$_{r=3}$	73.75	**83.80**	90.62	83.80	82.99
PromptFL	77.50	83.22	**90.97**	85.48	**84.29**
Ours	**78.76**	76.98	90.64	**88.11**	83.62
	C	P	R	A	
FedCLIP	68.61	87.37	88.06	78.00	80.51
FedProx	64.38	79.14	78.76	65.60	71.97
FedAVG	64.38	79.14	78.76	65.60	71.97
MOON	69.87	78.02	79.56	65.60	73.26
LoRA$_{r=3}$	**84.26**	90.74	86.34	69.96	82.83
PromptFL	82.75	**92.13**	87.73	71.88	83.62
Ours	76.98	90.87	**88.86**	**78.20**	**84.49**
	A	P	R	C	
FedCLIP	78.97	87.60	87.60	63.69	79.46
FedProx	73.81	80.38	80.48	57.64	73.08
FedAVG	73.81	80.38	80.48	57.64	73.08
MOON	70.93	80.38	80.83	52.94	71.27
LoRA$_{r=3}$	76.04	90.51	86.57	62.71	78.96
PromptFL	78.75	**92.71**	87.27	**66.66**	81.35
Ours	**81.44**	91.65	**90.58**	66.39	**82.51**
	A	C	R	P	
FedCLIP	78.56	68.50	87.37	87.52	80.49
FedProx	69.07	66.21	77.79	71.64	71.18
FedAVG	69.07	66.21	77.79	71.64	71.18
MOON	71.55	67.70	79.33	71.21	72.45
LoRA$_{r=3}$	75.62	**84.49**	86.00	85.42	82.88
PromptFL	77.71	80.79	88.31	85.19	83.0
Ours	**81.03**	75.49	**90.81**	**89.34**	**84.29**

approach are comparable with FedCLIP ($\sim 5 \times 10^5$), while is $3\times$ lower than LoRA ($\sim 1.5 \times 10^6$).

Results for OfficeHome. Table III reports the best testing accuracy, including baselines and our method for the Office-Home dataset. As can be seen, our method achieved the best average performance with an improvement of 1.29% (target is R) compared with the second-best method (PromptFL). Improvements are particularly notable for domain R.

Results for Multi-OF. Table VI reports the accuracy of the

TABLE IV: Accuracy(%) in the ModernOffice31 dataset. **Bold** means the best.

| | Source | | | Target | |
	A	D	S	W	Avg
FedCLIP	91.82	94.94	65.00	88.42	85.04
FedProx	82.41	81.82	74.03	79.49	79.43
FedAVG	59.07	69.38	59.67	64.53	63.16
MOON	84.01	81.82	74.68	83.14	80.91
LoRA$_{r=3}$	94.32	94.95	**93.23**	91.54	**93.51**
PromptFL	90.05	78.79	79.52	68.23	79.15
Ours	**95.38**	**98.99**	82.74	**91.95**	92.26
	D	S	W	A	
FedCLIP	95.95	65.32	88.05	90.06	84.84
FedProx	89.90	59.52	90.57	66.88	76.71
FedAVG	88.89	49.19	91.19	42.81	68.02
MOON	95.96	65.81	91.19	66.92	79.97
LoRA$_{r=3}$	97.98	**93.87**	**98.74**	76.28	**91.72**
PromptFL	81.82	76.13	83.65	68.93	77.63
Ours	**100**	80.32	94.34	**92.19**	91.71
	S	W	A	D	
FedCLIP	64.67	86.16	91.82	89.15	82.95
FedProx	73.55	90.57	81.53	89.36	83.75
FedAVG	76.94	93.08	82.59	92.17	86.19
MOON	75.00	92.45	87.39	85.74	85.14
LoRA$_{r=3}$	**92.58**	**96.23**	93.43	93.54	**93.95**
PromptFL	79.68	77.99	90.41	67.71	78.95
Ours	81.93	93.71	**95.38**	**95.98**	91.75
	W	A	D	S	
FedCLIP	87.42	91.47	95.95	54.77	82.40
FedProx	91.82	81.17	89.90	46.45	77.33
FedAVG	95.60	80.46	97.98	45.87	79.97
MOON	94.34	82.06	89.90	54.77	80.27
LoRA$_{r=3}$	**97.48**	**95.56**	**98.99**	48.93	85.24
PromptFL	81.76	88.45	85.86	32.49	72.14
Ours	96.23	94.85	**98.99**	**57.10**	**86.79**

TABLE V: BACC (%) and F1 score (%) in SC and HK datasets. **Bold** means the best.

| | | Source | | | | | | Target | | | |
		C_1		C_2		C_3		C_{glo}		Avg	
SC	FedCLIP	56.79	53.52	51.98	40.21	56.33	41.91	53.74	41.09	54.71	44.18
	FedProx	54.46	50.09	49.51	40.29	55.75	42.41	50.63	40.50	52.59	43.32
	FedAVG	55.93	50.38	50.61	39.33	56.82	43.95	51.38	39.97	53.68	43.41
	MOON	56.22	51.38	51.79	39.84	56.73	43.97	50.78	39.75	53.88	43.73
	LoRA$_{r=3}$	**83.73**	**81.34**	**72.05**	**63.98**	**81.71**	**71.96**	**74.23**	**63.54**	**77.93**	**70.2**
	PromptFL	60.43	61.98	53.93	54.72	59.94	59.91	59.88	59.50	58.55	59.03
	Ours	82.14	79.01	70.95	58.60	81.63	66.35	73.15	61.22	76.97	66.29
HK	FedCLIP	50.86	45.46	53.03	45.26	56.33	45.04	49.62	46.73	51.03	45.62
	FedProx	57.13	51.06	56.24	52.22	57.35	53.09	52.66	48.33	55.84	51.17
	FedAVG	56.05	50.63	57.33	51.71	56.51	52.83	53.01	50.11	55.72	51.32
	MOON	55.94	51.82	56.53	52.68	57.70	52.55	54.65	50.73	56.20	51.94
	LoRA$_{r=3}$	60.08	**57.29**	**60.45**	**56.20**	**59.14**	**56.90**	56.23	55.17	**58.98**	**56.39**
	PromptFL	48.35	47.78	45.70	44.39	47.25	45.72	44.74	42.77	46.51	45.16
	Ours	**62.40**	53.59	58.02	50.49	57.38	50.51	53.48	49.51	57.82	51.02

test sets for the baselines and our method. As illustrated, fine-tuning the entire CLIP model can achieve feasible test ACC in local clients (e.g., 83.85% ACC using FedProx in C_6), while indicates lower ACC in unseen domain (e.g., 83.48% accuracy in domain R using FedAVG). Concerning training efficient methods such as LoRA, PromptFL, FedCLIP and ours, the use of deeper FAM and DA in FAA-CLIP can lead to a higher ACC test in domain R (87.52%) compared to LoRA (81.23%) and PromptFL (44.55%). Surprisingly, FAA-CLIP provides the highest test ACC in C_{11} to C_{15}, demonstrating the potential of FAA-CLIP in FL.

Results for ModernOffice31. As reported in Table IV,

FAA-CLIP also achieves comparable performance compared to LoRA in {ADSW} (92.26% Avg vs 93.51% Avg) and {DSWA} (91.71% Avg vs 91.72% Avg), while indicates the highest average test ACC (86.79%) in {WADS} for the ModernOffice31 dataset. Notably, FAA-CLIP gives remarkable performance in domain S, showing its potential for synthetic data.

Results for BT. Once more, our method obtains the best average performance (88.18%) for the BT dataset (Figure 4). In particular, our method reaches a test ACC above 80% in C_{glo}, outperforming the second-best FL approach by over $\sim$ 10%.

Results for SC. The performance of the compared approaches for the SC dataset is summarized in Figure 4 and Table V. For ACC and BACC, FAA-CLIP outperforms FedCLIP by over 10%. Furthermore, our method achieves an F1 score near 61%, largely outperforming approaches like FedCLIP which only reach about 41%. Moreover, a prompt-tuning technique such as PromptFL indicates less effective compared to FAA-CLIP in skin cancer classification (e.g., 71.18% AVG vs 77.37% AVG for ACC). In addition, FAA-CLIP can provide similar performance compared to LoRA with less communication costs as reported in Table II. Figure 3 shows the test ACC, BACC and macro F1 for different communication epochs in each client. As illustrated, FAA-CLIP achieves a fast convergence, reaching a high classification ACC in a few epochs.

Results for HK. As reported in Figure 4 and Table V, our method achieves an ACC, BACC and macro F1 score comparable to MOON and FedProx for this dataset, while having lower resource costs than these approaches. Our approach outperforms FedCLIP by a large margin (>8%) in ACC. However, due to the class imbalance in the HK data set, FAA-CLIP indicates a lower performance (e.g., BACC) compared to LoRA (57.82% Avg vs 58.98% Avg).

AUC-ROC analysis. Figure 5 shows the receiver operator characteristics (ROC) curve with corresponding area under the curve (AUC) values for SC dataset. For simplicity, we used the one-vs-all method to measure the ROC and AUC and report the averaging curves/values for all classes. These curves show the more robust performance of our method for the medical datasets used in experiments.

Decision curve analysis. We further validate our method for clinical practice using a decision curve analysis (DCA) [43] on SC dataset. Typically, the best prediction to maximize net benefit is the one with the highest curve at any given threshold probability. Our experimental datasets are multi-class, thus we verify our method against existing FL methods using averaged net benefits (the average value of each class). As can be seen in Figure 5, our method provides a greater net benefit than other models, especially in the Client$_1$. This finding is consistent with the higher AUC-ROC values obtained by our approach.

Calibration analysis. For each sample of a specific class, the prediction model gives a confidence score that reflects the actual probability of the sample belonging to that class [44]. For example, calibration of the classification model ensures that the predicted probabilities match the actual probabilities.

TABLE VI: Accuracy (%) of test samples of OfficeHome based on the Dirichlet distribution (non-IID). The client domains are A (C_1 to C_5), C (C_6 to C_{10}) and P (C_{11} to C_{15}), while the global domain is R. **Bold** means the best.

|	Source															Target	AVG
	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	R	
FedAVG	72.92	83.33	80.21	81.25	75.00	82.81	84.38	80.21	**88.28**	82.50	88.75	89.06	**92.50**	83.59	92.97	83.48	83.83
FedProx	73.96	**85.42**	**81.25**	**83.33**	**78.12**	**83.85**	**85.42**	**85.42**	**88.28**	**85.62**	88.75	90.10	91.88	85.16	92.19	83.36	**85.13**
MOON	**76.04**	83.33	79.17	**83.33**	75.00	81.77	**85.42**	79.17	86.72	83.12	88.12	86.98	91.25	85.94	95.31	83.07	83.98
LoRA$_{r=3}$	65.62	**85.42**	72.92	77.08	67.19	80.73	83.85	78.12	**88.28**	79.38	88.75	94.79	91.88	84.38	**96.09**	81.23	82.23
PromptFL	53.12	82.29	71.88	69.79	60.94	69.27	67.19	61.46	81.25	74.38	86.88	84.38	88.12	78.91	87.50	44.55	72.62
FedCLIP	71.88	83.33	77.08	80.21	75.00	69.79	71.35	67.71	78.12	73.75	86.88	86.98	**92.50**	78.91	93.75	87.04	79.64
Ours	71.88	**85.42**	78.12	**83.33**	76.56	77.60	74.48	70.83	80.47	75.62	**90.62**	91.15	**92.50**	86.72	**96.09**	**87.52**	82.43
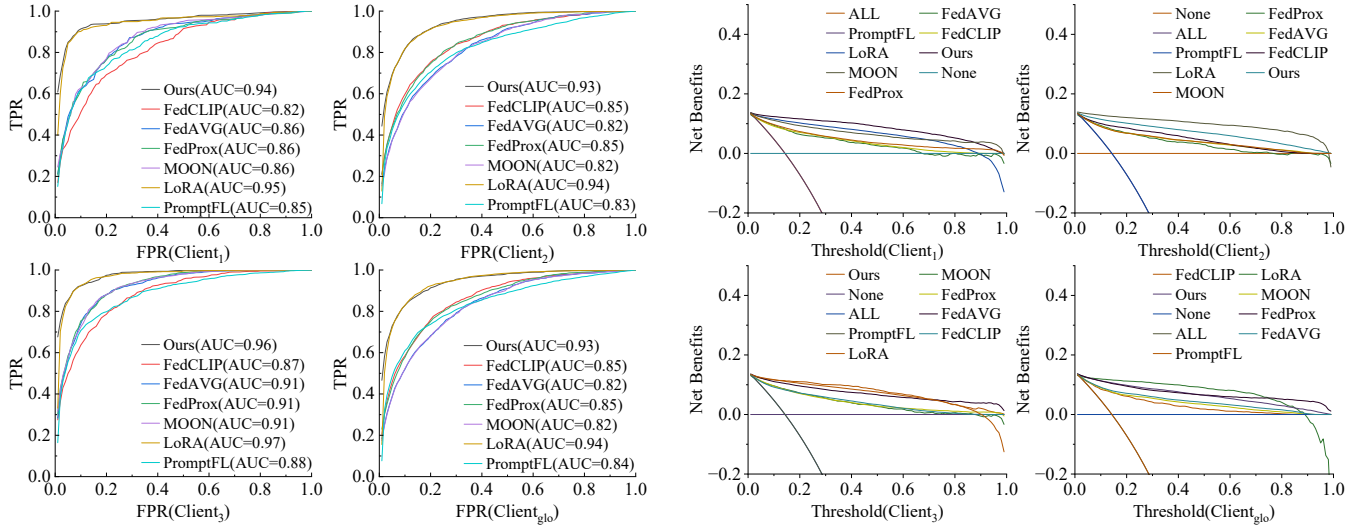
Fig. 5: The ROC curves and DCA of FAA-CLIP, FedCLIP, FedAVG, FedProx and MOON in skin cancer (SC) dataset. The ROC can measure the performance of classifiers, while DCA can assess the net benefit for clinical practices. The **left** part is the ROC curves, while the **right** part is the DCA.

TABLE VII: Accuracy (%) in OfficeHome, BT and SC. **Bold** means the best.

OfficeHome	Source			Target	Avg
	A	C	P	R	
Ours	78.76	76.98	90.64	88.11	83.62
w/ deeper FAM	**80.00**	**79.04**	90.98	88.09	**84.53**
w/o shared BN	78.76	76.97	90.64	88.11	83.62
w/ shared D	78.97	76.74	90.64	**88.16**	83.63
BT	C_1	C_2	C_3	C_{glo}	
Ours	**91.16**	90.47	89.36	**81.73**	**88.18**
w/ deeper FAM	87.07	89.28	89.94	79.44	86.43
w/o shared BN	85.71	89.28	85.18	78.42	84.65
w/ shared D	90.47	**90.47**	89.36	80.20	87.62
SC	C_1	C_2	C_3	C_{glo}	
w/o DA	79.94	71.49	81.02	71.50	75.99

set compared to FedCLIP, highlighting its potential for the medical domain.

Following [44], the expected calibration error (ECE) metric is used to evaluate the calibration effects. Figure 8 and Figure 10 illustrate the reliability plot with the ECE value for FAA-CLIP and FedCLIP using the OfficeHome ({ACPR}) and SC datasets. For natural images, FAA-CLIP provides a lower ECE value in the local client (3.12% vs 4.13%) while maintaining a similar ECE value in the global compared to FedCLIP (1.16% vs 0.59%). However, for medical images, FAA-CLIP reduces the ECE value $\sim 11\%$ on the local client and $\sim 2\%$ on global
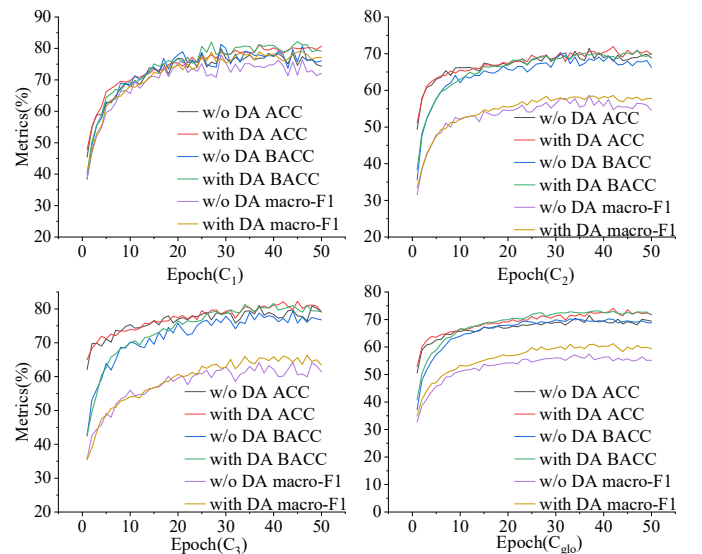

Fig. 6: Testing accuracy (ACC), balanced accuracy (BACC) and macro-F1 for each epoch using SC for FAA-CLIP (with DA and without (w/o) DA).

TABLE VIII: Accuracy (%) in OfficeHome and BT dataset. **Bold** means the best.

	Attention module layer					OfficeHome				BT				SC			
	L1	L2	L3	L4	L5	A	C	P	R	C_1	C_2	C_3	C_{glo}	C_1	C_2	C_3	C_{glo}
fea_1	✓	✗	✗	✗	✓	79.38	73.38	89.51	87.35	77.51	90.47	81.91	64.21	62.43	57.35	66.62	59.21
fea_2	✓	✓	✗	✗	✓	78.96	78.69	92.78	**88.50**	87.75	88.09	87.23	80.71	80.96	70.60	81.19	73.01
fea_3	✓	✓	✓	✗	✓	79.38	**81.09**	**91.77**	87.33	86.39	85.71	85.11	80.96	79.94	71.64	81.96	**74.15**
fea_4	✓	✗	✓	✓	✓	**80.20**	73.08	89.74	88.38	83.67	**92.85**	82.97	63.45	59.39	56.76	66.45	58.24
fea_5	✓	✓	✓	✓	✓	78.76	76.98	90.64	88.11	**91.16**	90.47	**89.36**	**81.73**	**81.22**	**71.95**	**82.26**	74.07

L1: Linear layer; L2: Batch normalization layer; L3: LeakyReLU layer; L4: Linear layer; L5: Softmax layer.

D. Ablation studies

We perform several ablation studies to evaluate the different components of our method. Except for the specific modules, backbones, and parameters described in the following, the configuration of our model is as described in Section IV-B.

Impact of the domain adaptation for homogeneous client. As illustrated in Table III, the combination of FAM and DA can provide feasible ACC for natural image classification when the client data come from different domains. One concern suggested in [45] is that if client data are from the same domain (e.g., the contribution of DA is less), can FAA-CLIP still exhibit remarkable ACC compared to SOTA methods? Thus, the OfficeHome is used for experiments. For a specific domain (e.g., C), following [20], practical and pathological heterogeneity are adopted to allocate data for each client. For practical purposes, data are partitioned into clients based on the Dirichlet distribution. During pathological examination, each client holds C/N (C: Number of classes, N: Number of clients) non-overlapping classes. Then, a different domain (e.g., R) is used for global model evaluation. In our experiments, we use the scikit-learn library to generate Dirichlet distribution randomly to allocate the data, and the number of clients is set to 5 (client domain is R, global domain is C). For pathological, each client holds 13 different classes (derived from domain R), while the global server has three domain data (ACP). Figure 9 shows the data distribution in each client, while Table IX reports the ACC test for the FAA-CLIP and SOTA methods. As illustrated, the usefulness of DA is limited as FAA-CLIP indicates a similar ACC testing compared to FedCLIP (e.g., 82.69% AVG vs 82.55% AVG). However, FAA-CLIP still outperforms LoRA with large margins (e.g., 82.69% AVG vs 80.54% AVG). For pathological, the use of DA demonstrates lower effects for out-of-distribution data (e.g., 73.54% ACC in domain {ACP}), while it can align local data (e.g., 95.16% ACC in C_3).

Influence of domain adaptation module. We validate the usefulness of DA on the SC dataset. Table VII reports the best testing ACC for each client. Furthermore, Figure 6 shows the testing ACC, BACC and F1 of each client for each communication round. As illustrated, model without DA achieves a lower performance compared to the one using DA. This demonstrates the need for DA to minimize data distribution shifts across clients and global testing set.

Impact FAM architecture. We measured our model's performance when removing different layers within the FAM (the first layer is defined as L1, and so on) to examine the effectiveness of these layers. We used the OfficeHome (Source {A,C,P}, Target R), BT and SC datasets for this experiment, and report results in Table VIII. Removing the batch normalization layer (L2) leads to a performance drop for C_{glo} ACC in the BT and SC datasets. In the OfficeHome dataset, which contains natural images, the impact of batch normalization layers is less significant, and having a larger number of linear layers may result in a performance degradation. It is thus imperative to determine the optimal number of layers.

Impact of FAM depth. We examine the impact of having a more complex feature adaptation module on performance and efficiency. Thus, we change the FAM to have 8 layers instead of 5: a first linear layer, a batch normalization layer, a LeakyReLU layer, a second linear layer, a batch normalization layer, a LeakyReLU layer, a third linear layer and a softmax activation function. We use the OfficeHome and BT dataset for testing. Table VII reports the highest test ACC for these datasets. As can be seen, the new FAM achieves almost identical average performance compared to the original results. While the performance is higher in domain C, the number of parameters of this deeper FAM is nearly $1.5\times$ larger than the original one, thereby increasing communication costs.

Impact of batch normalization. Experimental results in [15] have shown the effectiveness of keeping local the batch normalization parameters of each client (not aggregating them in the server) in the context of CNN-based classification. To verify these results, we updated the batch normalization parameters of each client's FAM independently, and used OfficeHome and BT data sets to measure test performance. Table VII gives the highest test ACC for this ablation experiment. As reported, keeping the batch normalization parameters local leads to a worse performance than sharing and aggregating these parameters. This surprising result suggests that local batch normalization is less helpful for FL with VLMs like CLIP.

Impact of sharing domain classifier parameters. We studied the influence of aggregating domain classifier D parameters on model performance. Table VII reports the highest test ACC using the OfficeHome and BT datasets. By sharing the parameters of the domain classifier, we gain in performance for domains A and R in OfficeHome. However, for BT, we observe an identical or lower performance compared with the original results.

Impact of backbone. We replaced the ViT-B/32 backbone of the image encoder with pretrained deeper architectures, ViT-L/14 and ViT-L/14@336px, to verify the effectiveness of FAA-
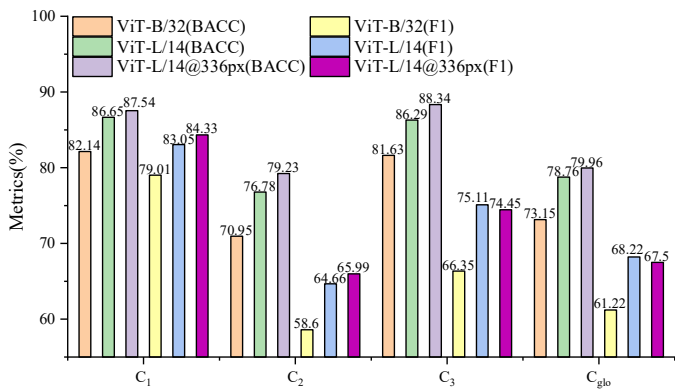
Fig. 7: Testing BACC and macro-F1 of FAA-CLIP for SC dataset, with the ViT-B/32, ViT-L/14 and ViT-L/14@336px backbones.

TABLE IX: Testing accuracy (%) in OfficeHome dataset with non-IID under practical and pathological settings. **Bold** means the best.

R → C		C_1	C_2	C_3	C_4	C_5	C	AVG
Practical	FedAVG	82.72	87.50	**87.21**	81.77	80.12	60.11	79.91
	FedProx	79.01	86.31	**87.21**	85.08	81.33	58.16	79.52
	MOON	85.80	**88.10**	84.88	80.66	83.73	60.78	80.66
	FedCLIP	81.48	86.90	83.72	**92.27**	87.95	62.98	82.55
	LoRA	**83.33**	85.71	**87.21**	86.74	85.54	54.69	80.54
	PromptFL	38.89	52.38	49.42	48.62	47.59	14.45	41.89
	Ours	82.10	86.90	84.30	91.71	**87.95**	**63.21**	**82.69**

R → ACP		C_1	C_2	C_3	C_4	C_5	ACP	AVG
Pathological	FedAVG	95.03	96.17	93.55	90.45	89.44	64.28	88.15
	FedProx	**96.89**	**96.72**	**95.70**	96.63	86.96	**75.32**	91.37
	MOON	96.27	96.17	93.55	91.01	91.93	66.41	89.22
	FedCLIP	96.27	**96.72**	94.09	97.19	91.93	74.04	91.71
	LoRA	94.41	93.44	94.09	93.26	**93.79**	54.27	87.21
	PromptFL	85.09	89.62	87.63	85.39	86.34	18.89	75.49
	Ours	**96.89**	96.17	95.16	**97.75**	**93.79**	73.54	**92.22**

CLIP. We used the SC dataset for these experiments. Figure 7 shows the best test BACC and macro-F1 for each client and the global server. While it incurs the same communication cost, employing a larger backbones like ViT-L/14 or ViT-L/14@336px leads to a better BACC and F1 score for all clients.

V. DISCUSSIONS

As reported in [46], our results in Table III and Figure 4 indicate the poor performance of standard VLM fine-tuning in FL. Instead of fine-tuning the whole model, our FAA-CLIP employs a light-weight adaptation technique based on the proposed FAM. This technique significantly reduces computational needs (e.g., our FAM only needs to transmit nearly 10^5 parameters) in the case of VLMs like CLIP [46]. The proposed FAM only needs 2.5MB of memory to store the parameters, which is nearly $140\times$ less than the CLIP encoders with 345MB. Similar to FedCLIP, our FAM is a plug-and-play technique that can be easily extended to other vision foundation models or vision language models. As demonstrated in Section IV-D, the proposed FAM can substantially improve performance in medical imaging datasets, thus bridging the gap with the more common natural image datasets. Furthermore, while sharing FAM paramters can alleviate the problem
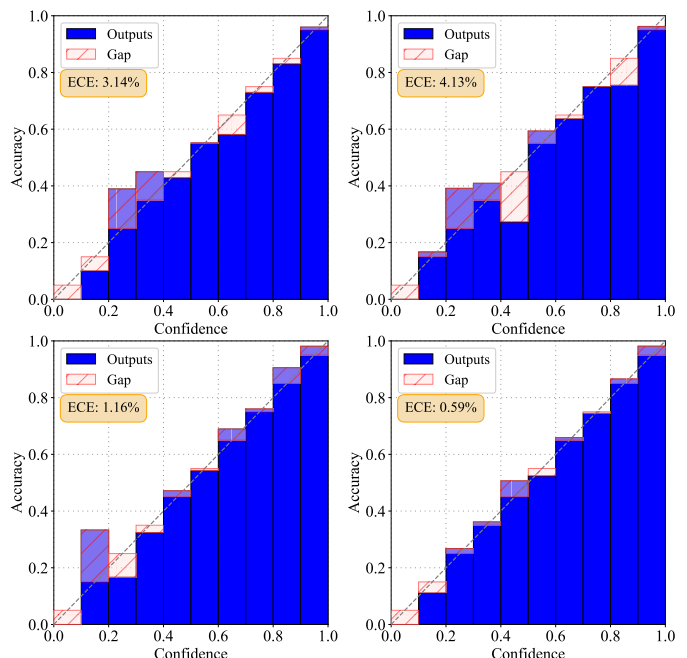

Fig. 8: Reliability diagram for FAA-CLIP (**left** column) vs FedCLIP (**right** column) using the OfficeHome ({ACPR}) dataset. The first row represents the client A, while the second row indicates the global R.
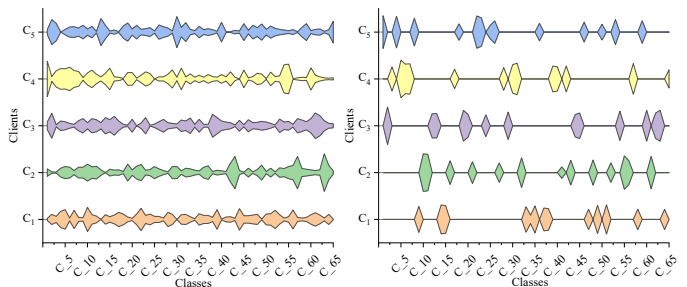

Fig. 9: Example of data distribution in each client using kite graph in OfficeHome dataset under practical and pathological setting. C_1 to C_5 indicate clients while C_1 to C_65 represent classes.

of domain shifts among clients, using an explicit strategy for DA can further improve the performance of models on unseen domains (e.g., C_{glo} in SC). As stated in Section IV-D, it is also important to maintain a balance between communication costs and parameters. Furthermore, the findings of ROC-AUC and DCA described in Secion IV-C suggest that our method can achieve competitive AUC values with high net benefits, highlighting its potential for clinical applications.

When applied to larger models such as ViT-L/14, our approach achieves a classification F1 score that exceeds 72% in a large-scale SC classification dataset, thereby providing practical clinical value. However, as reported in our results, sharing the domain classifier D can cause the local model to lose its learned feature representation, thus reducing its performance. This suggests that not all components are suitable for global aggregation; instead, it is recommended to choose the best strategy for aggregation. Experiments on the HK dataset indicate that our method can achieve a comparable
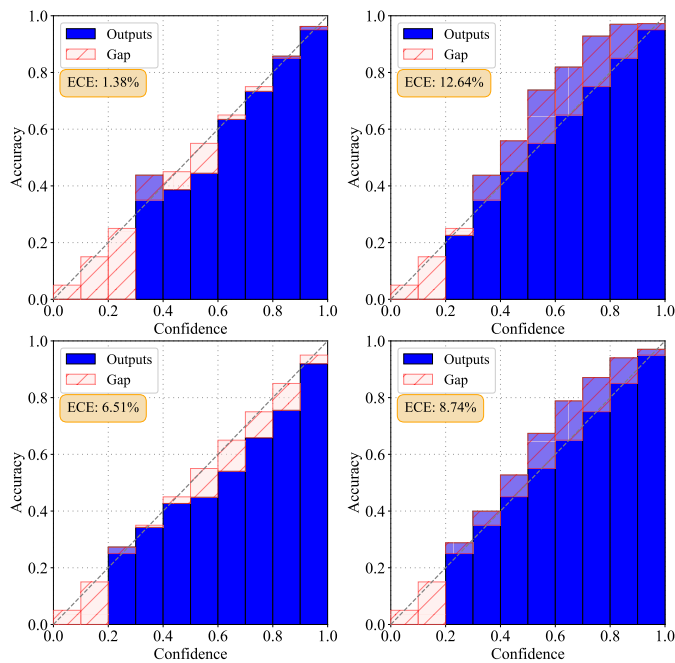
Fig. 10: Reliability diagram for FAA-CLIP (**left** column) vs FedCLIP (**right** column) on SC dataset. The first row represents the client C_2, while the second row indicates the global C_{glo}.

performance compared to fine-tuning the entire CLIP model, with less computational costs. This promotes the development of FL methods for gastrointestinal endoscopy. In addition, FAA-CLIP, while not originally designed for calibration tasks, shows potential for them by achieving a lower ECE value compared to other SOTA methods. This indicates that the FAM itself can be applied to various tasks beyond its initial design.

For future work, we intend to integrate FAA-CLIP into other frameworks and investigate the benefit of having a separate FAM for the text encoder. Exploring more powerful adaptation methods could also improve the performance of our FAA-CLIP model. In what follows, we highlight the strengths and limitations of our approach.

Strengths and limitations. Our approach is easy to extend (e.g., FAM is easy to customize), and communication efficient compared with traditional FL methods. Furthermore, our method can consistently outperform other FL approaches in commonly used datasets. It also has a faster convergence rate (e.g., Figure 3 shows our method can achieve the same performance with less communication epoch in SC) in medical datasets compared to alternative approaches, demonstrating its effectiveness.

This study used a simple yet effective technique for the feature adaptation. Nevertheless, edge devices like mobile phones may not be able to support the computational expenses related to feature adaptation in other applications such as remote monitoring [47]. We will explore more efficient DA techniques as suggested in [48]. Furthermore, for the class imbalance problem that occurs in the medical field, the ACC is not consistent with the F1 score (e.g., >10% performance drop for the F1 score). In the future, we will investigate a contrastive learning strategy specifically designed for rare diseases such

as those described in [49] to solve this problem. In our experiments, hyper-parameters of FAA-CLIP (e.g., learning rate) were fixed consistently in the local training stage. Using optimizer parameter adjustment (e.g., exponential learning rate regularization) to tune these parameters may help get a better performance.

VI. Conclusion

In this paper, we explored the usefulness of VLMs for image classification in the context of FL. To reduce communication costs in FL, we proposed a FAM adapter and only shared the parameters of this module. Furthermore, we introduced a domain adaptation strategy to reduce domain shifts among clients. Experimental results on three natural image datasets and three medical imaging datasets demonstrated the effectiveness of our method compared to state-of-the-art FL approaches. In addition, we conducted a series of ablation studies on various components and design choices, including the FAM architecture and the influence of DA, to better guide researchers in building FL frameworks with VLMs. Our study can help bridge the gap in performance of VLMs in FL scenarios related to medical imaging.

Acknowledgments

References

[1] T. V. Khoa, D. T. Hoang, N. L. Trung, C. T. Nguyen, T. T. T. Quynh, D. N. Nguyen, N. V. Ha, and E. Dutkiewicz, "Deep transfer learning: A novel collaborative learning model for cyberattack detection systems in iot networks," *IEEE Internet of Things Journal*, vol. 10, no. 10, pp. 8578–8589, 2023.

[2] P. Voigt and A. Von dem Bussche, "The eu general data protection regulation (gdpr)," *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, vol. 10, no. 3152676, pp. 10–5555, 2017.

[3] H. Rafi, Y. Benezeth, F. Y. Song, P. Reynaud, E. Arnoux, and C. Demonceaux, "Tree-based personalized clustered federated learning: A driver stress monitoring through physiological data case study," *IEEE Internet of Things Journal*, pp. 1–1, 2024.

[4] C. You, K. Guo, G. Feng, P. Yang, and T. Q. S. Quek, "Automated federated learning in mobile-edge networks—fast adaptation and convergence," *IEEE Internet of Things Journal*, vol. 10, no. 15, pp. 13571–13586, 2023.

[5] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*, pp. 1273–1282, PMLR, 2017.

[6] A. Chaddad, Y. Wu, and C. Desrosiers, "Federated learning for healthcare applications," *IEEE Internet of Things Journal*, vol. 11, no. 5, pp. 7339–7358, 2024.

[7] A. Chaddad and Y. Wu, "Enhancing classification tasks through domain adaptation strategies," in *2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 1832–1835, IEEE, 2023.

[8] A. Mora, A. Bujari, and P. Bellavista, "Enhancing generalization in federated learning with heterogeneous data: A comparative literature review," *Future Generation Computer Systems*, 2024.

[9] J. P. Huix, A. R. Ganeshan, J. F. Haslum, M. Söderberg, C. Matsoukas, and K. Smith, "Are natural domain foundation models useful for medical image classification?," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 7634–7643, 2024.

[10] Y. Wu, C. Desrosiers, and A. Chaddad, "Facmic: Federated adaptative clip model for medical image classification," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 531–541, Springer, 2024.

[11] W. Lu, X. HU, J. Wang, and X. Xie, "Fedclip: Fast Generalization and Personalization for CLIP in Federated Learning," in *ICLR 2023 Workshop on Trustworthy and Reliable Large-Scale Machine Learning Models*, 2023.

[12] J. Lee, F. Solat, T. Y. Kim, and H. V. Poor, "Federated learning-empowered mobile network management for 5g and beyond networks: From access to core," *IEEE Communications Surveys & Tutorials*, 2024.

[13] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-iid data," *IEEE transactions on neural networks and learning systems*, vol. 31, no. 9, pp. 3400–3413, 2019.

[14] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems*, vol. 2, pp. 429–450, 2020.

[15] X. Li, M. Jiang, X. Zhang, M. Kamp, and Q. Dou, "Fedbn: Federated learning on non-iid features via local batch normalization," *arXiv preprint arXiv:2102.07623*, 2021.

[16] Q. Li, B. He, and D. Song, "Model-contrastive federated learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10713–10722, 2021.

[17] H. Yuan, W. R. Morningstar, L. Ning, and K. Singhal, "What do we mean by generalization in federated learning?," in *International Conference on Learning Representations*, 2022.

[18] Z. Qu, X. Li, R. Duan, Y. Liu, B. Tang, and Z. Lu, "Generalized federated learning via sharpness aware minimization," in *International conference on machine learning*, pp. 18250–18280, PMLR, 2022.

[19] R. Zhang, Q. Xu, J. Yao, Y. Zhang, Q. Tian, and Y. Wang, "Federated domain generalization with generalization adjustment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3954–3963, 2023.

[20] Z. Qin, S. Deng, M. Zhao, and X. Yan, "Fedapen: personalized cross-silo federated learning with adaptability to statistical heterogeneity," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1954–1964, 2023.

[21] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*, pp. 8748–8763, PMLR, 2021.

[22] L. Fan, D. Krishnan, P. Isola, D. Katabi, and Y. Tian, "Improving clip training with language rewrites," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[23] T. Guo, S. Guo, J. Wang, X. Tang, and W. Xu, "Promptfl: Let federated participants cooperatively learn prompts instead of models-federated learning in age of foundation model," *IEEE Transactions on Mobile Computing*, 2023.

[24] M. Singha, H. Pal, A. Jha, and B. Banerjee, "Ad-clip: Adapting domains in prompt space using clip," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4355–4364, 2023.

[25] J. Shi, S. Zheng, X. Yin, Y. Lu, Y. Xie, and Y. Qu, "Clip-guided federated learning on heterogeneity and long-tailed data," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, pp. 14955–14963, 2024.

[26] M. Zanella and I. Ben Ayed, "Low-rank few-shot adaptation of vision-language models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1593–1603, 2024.

[27] S. Chen, J. Gu, Z. Han, Y. Ma, P. Torr, and V. Tresp, "Benchmarking robustness of adaptation methods on pre-trained vision-language models," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[28] T. Guo, S. Guo, and J. Wang, "Pfedprompt: Learning personalized prompt for vision-language models in federated learning," in *Proceedings of the ACM Web Conference 2023*, pp. 1364–1374, 2023.

[29] H. Chen, Y. Zhang, D. Krompass, J. Gu, and V. Tresp, "Feddat: An approach for foundation model finetuning in multi-modal heterogeneous federated learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, pp. 11285–11293, 2024.

[30] W. Tu, W. Deng, and T. Gedeon, "A closer look at the robustness of contrastive language-image pre-training (clip)," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[31] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. March, and V. Lempitsky, "Domain-adversarial training of neural networks," *Journal of machine learning research*, vol. 17, no. 59, pp. 1–35, 2016.

[32] C. Sudlow, J. Gallacher, N. Allen, V. Beral, P. Burton, J. Danesh, P. Downey, P. Elliott, J. Green, M. Landray, *et al.*, "Uk biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age," *PLoS medicine*, vol. 12, no. 3, p. e1001779, 2015.

[33] D. C. Van Essen, K. Ugurbil, E. Auerbach, D. Barch, T. E. Behrens, R. Bucholz, A. Chang, L. Chen, M. Corbetta, S. W. Curtiss, *et al.*, "The human connectome project: a data acquisition perspective," *Neuroimage*, vol. 62, no. 4, pp. 2222–2231, 2012.

[34] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5018–5027, 2017.

[35] T. Ringwald and R. Stiefelhagen, "Adaptiope: A modern benchmark for unsupervised domain adaptation," in *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 101–110, 2021.

[36] P. Tschandl, C. Rosendahl, and H. Kittler, "The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Scientific data*, vol. 5, no. 1, pp. 1–9, 2018.

[37] N. C. Codella, D. Gutman, M. E. Celebi, B. Helba, M. A. Marchetti, S. W. Dusza, A. Kalloo, K. Liopyris, N. Mishra, H. Kittler, *et al.*, "Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic)," in *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pp. 168–172, IEEE, 2018.

[38] M. Combalia, N. C. Codella, V. Rotemberg, B. Helba, V. Vilaplana, O. Reiter, C. Carrera, A. Barreiro, A. C. Halpern, S. Puig, *et al.*, "Bcn20000: Dermoscopic lesions in the wild," *arXiv preprint arXiv:1908.02288*, 2019.

[39] H. Borgli, V. Thambawita, P. H. Smedsrud, S. Hicks, D. Jha, S. L. Eskeland, K. R. Randel, K. Pogorelov, M. Lux, D. T. D. Nguyen, D. Johansen, C. Griwodz, H. K. Stensland, E. Garcia-Ceja, P. T. Schmidt, H. L. Hammer, M. A. Riegler, P. Halvorsen, and T. de Lange, "HyperKvasir, a comprehensive multi-class image and video dataset for gastrointestinal endoscopy," *Scientific Data*, vol. 7, no. 1, p. 283, 2020.

[40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[42] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.

[43] A. J. Vickers and E. B. Elkin, "Decision curve analysis: a novel method for evaluating prediction models," *Medical Decision Making*, vol. 26, no. 6, pp. 565–574, 2006.

[44] B. Murugesan, J. Silva-Rodríguez, I. B. Ayed, and J. Dolz, "Robust calibration of large vision-language adapters," in *European Conference on Computer Vision*, pp. 147–165, Springer, 2025.

[45] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE transactions on neural networks*, vol. 22, no. 2, pp. 199–210, 2010.

[46] P. Gao, S. Geng, R. Zhang, T. Ma, R. Fang, Y. Zhang, H. Li, and Y. Qiao, "Clip-adapter: Better vision-language models with feature adapters," *International Journal of Computer Vision*, vol. 132, no. 2, pp. 581–595, 2024.

[47] G. Longo, A. Orlich, A. Merlo, and E. Russo, "Enabling real-time remote monitoring of ships by lossless protocol transformations," *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[48] A. Chaddad, Y. Wu, Y. Jiang, A. Bouridane, and C. Desrosiers, "Simulations of common unsupervised domain adaptation algorithms for image classification," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 1–17, 2025.

[49] H. Zhang, W. Liu, Z. Li, J. Shi, S. Chang, H. Wang, J. He, and Q. Huang, "Sigxcl: A signal–image–graph cross-modal contrastive learning framework for cvd diagnosis based on internet of medical things," *IEEE Internet of Things Journal*, vol. 11, no. 7, pp. 12984–13001, 2024.