Bridging the Pathology Domain Gap: Efficiently Adapting CLIP for Pathology Image Analysis with Limited Labeled Data

Zhengfeng Lai[1], Joohi Chauhan[2], Brittany N. Dugger[1], and Chen-Nee Chuah[1]

[1] University of California, Davis
[2] Motilal Nehru National Institute of Technology Allahabad
{lzhengfeng, bndugger, chuah}@ucdavis.edu
joohi@mnnit.ac.in

Abstract. Contrastive Language-Image Pre-training (CLIP) has shown its proficiency in acquiring distinctive visual representations and exhibiting strong generalization across diverse vision tasks. However, its effectiveness in pathology image analysis, particularly with limited labeled data, remains an ongoing area of investigation due to challenges associated with significant domain shifts and catastrophic forgetting. Thus, it is imperative to devise efficient adaptation strategies in this domain to enable scalable analysis. In this study, we introduce Path-CLIP, a framework tailored for a swift adaptation of CLIP to various pathology tasks. Firstly, we propose Residual Feature Refinement (RFR) with a dynamically adjustable ratio to effectively integrate and balance source and task-specific knowledge. Secondly, we introduce Hidden Representation Perturbation (HRP) and Dual-view Vision Contrastive (DVC) techniques to mitigate overfitting issues. Finally, we present the Doublet Multimodal Contrastive Loss (DMCL) for fine-tuning CLIP for pathology tasks. We demonstrate that Path-CLIP adeptly adapts pre-trained CLIP to downstream pathology tasks, yielding competitive results. Specifically, Path-CLIP achieves over $+$**19%** improvement in accuracy when utilizing mere **0.1%** of labeled data in PCam with only 10 minutes of fine-tuning while running on a single GPU.

1 Introduction

Recent advancements in deep learning, bolstered by superior network architectures and the availability of extensive, meticulously curated datasets, have led to substantial enhancements in pathology image analysis tasks [17]. Nevertheless, the process of collecting high-quality datasets with reliable annotations for each specific vision task remains a laborious and time-intensive endeavor [18, 53]. This may prevent the broad adoption of advanced deep learning techniques. To relieve the reliance on such datasets, pre-training and fine-tuning methods (TFT) have been studied in vision tasks: pre-train the model on a large-scale dataset and then fine-tune the model on different downstream tasks [9]. There are several challenges of such methods: 1) they may still require a large amount of labeled

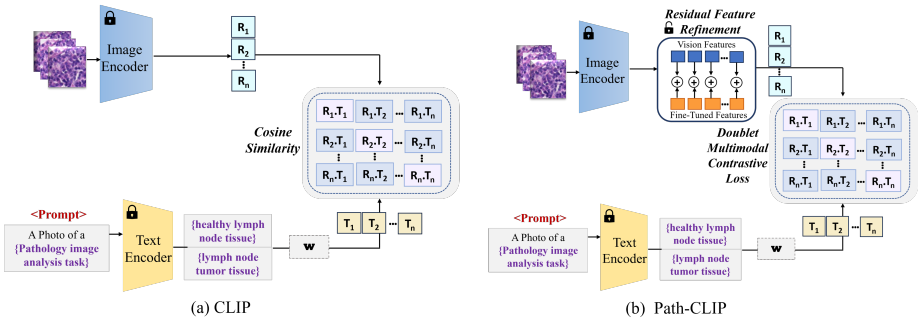


Fig. 1: Illustration of (a) zero-shot CLIP, and (b) our proposed Path-CLIP. We aim to freeze the encoders but only train the Residual Feature Refinement (RFR) module to balance the original knowledge from CLIP and new task-specific knowledge.

set to avoid the overfitting issue when fine-tuning the model for the downstream task [21, 40]; 2) the fine-tuning may not yield satisfactory performance in the target domain due to the existence of a large domain gap between the pre-trained data and pathology images [32]. Consequently, a pivotal question arises: how can the large-scale pre-trained model be efficiently tailored to a new task, minimizing overfitting, particularly when the new dataset is limited in size?

To fill the performance gap due to domain shift, Contrastive Language-Image Pre-training (CLIP) [26] has shown its power in learning generic and distinctive visual representations via language supervision. As shown in Fig. 1(a), it aligns images and texts in the same feature space and uses a contrastive loss to formulate the learning objective. CLIP uses two separate encoders for images and texts, then maximizes the similarity score of positive pairs of images and texts while minimizing it of negative pairs [26, 54]. It achieves promising results on various image classification tasks without needing any annotated data, i.e., zero-shot transfer settings. As a language-vision model, CLIP uses prompts as the supervision, where the visual labels are entered into the hand-crafted template. By pre-training the model at a large scale, models can learn the visual contents and easily be transferred to downstream tasks through the prompt-based zero-shot transfer.

However, the manual design of prompts can be a non-trivial and time-consuming task. [54] found that even a slight change in the prompt (e.g., one word) can make a big difference. They introduced Context Optimization (CoOp) to automate prompt engineering to generate continuous soft prompts instead of using hand-chosen hard prompts [54]. However, CoOp requires substantial computing resources, and the results of CoOp are not interpretable. Furthermore, such prompt tuning faces performance degradation when there is a significant domain shift [39], e.g., from natural images to pathology images, rendering it less adaptable to medical imaging tasks. On the other hand, medical image analysis may involve multiple learning tasks simultaneously. Conversely, in medical im-

age analysis, multiple learning tasks may be concurrently required. It becomes impractical if fine-tuning necessitates substantial computational resources. For instance, Lai et al. [17] quantify and visualize distinct pathologies in grey/white matter (GM/WM) regions within gigapixel images. To accomplish this, the task is divided into two learning objectives: pathology detection and GM/WM segmentation. They collected and annotated two datasets and then trained two models from scratch to achieve it. Therefore, it will be beneficial if we can propose an effective way to adapt a big model into multiple downstream tasks quickly so the final analysis objective can be reached in a reasonable time manner.

In this work, we aim to fine-tune CLIP efficiently with light computing resources for pathology image analysis tasks. There are several challenges in fine-tuning CLIP. First, **overfitting** is a severe issue if we adapt CLIP to the downstream tasks directly since CLIP is pre-trained on a 400M dataset while the new domain dataset can be small [10]. Second, CLIP has shown its promising zero-shot generalization ability, directly fine-tuning CLIP may lose this capability and face the "**catastrophic forgetting**" issue. In other words, it is unclear how to effectively learn the new knowledge while retaining the original pre-trained knowledge so that the generalization of CLIP can be maintained. Third, there are limited studies on benchmarking CLIP's transferability in the pathology domain; hence, its applicability remains unclear. To address the above issues, we first study the applicability of CLIP on two public pathology datasets and benchmark the zero-shot ability of CLIP on them. Then, we did empirical experiments to verify the importance of the CLIP's pre-trained knowledge. To address the "catastrophic forgetting" issue, we propose Residual Feature Refinement (RFR) as a lightweight approach for adapting CLIP to pathology images. As a shallow network, RFR demonstrates proficiency in extracting crucial spatial features and patterns from pathology images. Within RFR, we introduce a self-adaptive module designed to seamlessly fuse and balance the original knowledge from CLIP with the newly acquired insights from the specific pathological task, requiring only a minimal number of additional trainable weights, as opposed to retraining the entire encoders in CLIP. To further mitigate overfitting, we also incorporate Hidden Representation Perturbation (HRP) and Dual-view Vision Contrastive (DVC). Finally, we establish an end-to-end adaptation of CLIP, named Path-CLIP (as shown in Fig. 1(b)), tailored for pathology image analysis tasks. Path-CLIP incorporates a newly proposed loss function, Doublet Multimodal Contrastive Loss (DMCL).

We summarize our contributions as follows:

- We explore the applicability of CLIP on pathology images and highlight the catastrophic forgetting issues when adapting CLIP to downstream pathology tasks.

- We propose Path-CLIP to alleviate the overfitting and forgetting issues and effectively adapt CLIP for pathology tasks.

– Extensive experiments demonstrate the efficiency and robustness of Path-CLIP. For example, it achieves +**19%** on PCam when only using **0.1%** of labeled data.

2 Related Work

Language-vision model (LVM). LVMs have shown promising performance in learning generic visual representations [15, 22, 26, 48]. Recent models advance via text representation learning with large-scale Transformers [30] and even web-scale training datasets [54]. Transformer-based multimodal learning has achieved great success in such large-scale datasets [23, 29, 36]. For example, CLIP [26] was trained on 400 million image-caption pairs and achieved state-of-the-art results in many fields [26, 44, 46, 47]. In CLIP's architecture, it has two encoders: the vision encoder can be ResNet [12] or ViT [7] while the text encoder can be Transformer (e.g., BERT [6]). A recent work [27] fine-tuned CLIP for video data and performed competitively to more complicated methods designed for video processing. PointCLIP [50] applied CLIP to 3D recognition. CLIP is also applied for image generation tasks [8] and shows its power to reduce required data collection. However, the best way to adapt CLIP for downstream tasks is still being studied, especially when the new domain is far from the pre-trained domains, e.g., the medical domain.

CLIP Fine-tuning and adaptation. Although recent multimodal models (e.g., CLIP [26]) perform astonishingly well in standard vision tasks without fine-tuning, the gap between CLIP and supervised learning is still noticeable. Therefore, many works focus on fine-tuning CLIP with a few samples from the new task provided. For example, CoOp [54] kept the entire pre-trained parameters fixed but optimized the continuous prompt through a learnable vector. This method closes the gap between CLIP and supervised learning but the performance gain is limited as it only focuses on the perspective of prompt design instead of the vision part for the image recognition tasks. On the other hand, CLIP-Adapter [9] and Tip-Adapter [51] freeze the pre-trained weights and train additive module, which significantly improves the CLIP's performance with limited training data in the new task. These methods require a lower computational cost. However, both CLIP-Adapter [9] and Tip-Adapter [51] involve tuning a ratio (hyper-parameter) to balance the original and fine-tuned knowledge manually, which may not be practical when we have multiple target learning tasks in the medical domain.

3 Methodology

In this section, we introduce the proposed Path-CLIP. We summarize the end-to-end pipeline in Fig. 1(b): a self-adaptive residual ratio α will be incorporated into Residual Feature Refinement (Section 3.2) to fuse and balance the original knowledge with the task-specific knowledge from the new downstream task. We introduce Hidden Representation Perturbation (HRP) and Dual-view Vision

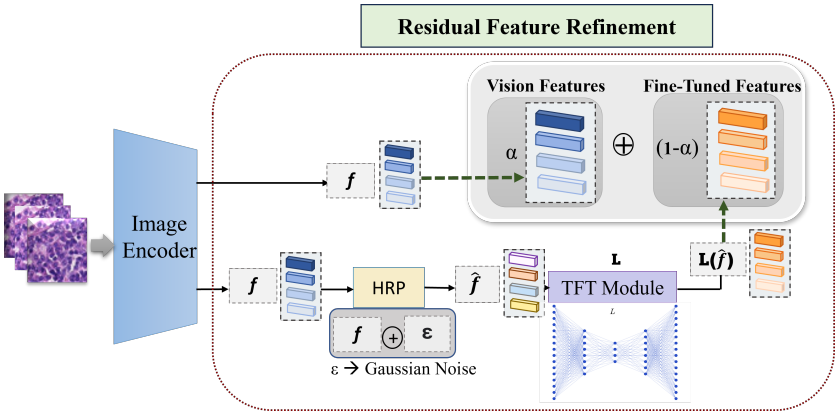


Fig. 2: Overview of the Residual Feature Refinement (RFR): a downsampling-upsampling shallow architecture of fully connected layers. RFR blends the fine-tuned knowledge with the original knowledge from CLIP's vision encoder ($F(\cdot)$) via a self-adaptive residual ratio α. HRP refers to Hidden Representation Perturbation and TFT refers to pre-training fine-tuning. We use blue color to denote the original visual features and orange color to denote the fine-tuned features.

Contrastive (Section 3.3) to alleviate the overfitting issue. Finally, we propose Doublet Multimodal Contrastive Loss (DMCL) in Section 3.4 as the loss function for fine-tuning.

3.1 Language-Vision Pre-training

We first revisit language-vision pre-training and use CLIP [26] as one example. Note that our framework is applicable to broader language-vision models. CLIP [26] has a vision encoder $F(\cdot)$ and a text encoder $G(\cdot)$. The vision encoder maps a high-dimensional image into low-dimensional image embeddings. The vision encoder can be a CNN like ResNet [12] or a Vision Transformer (ViT) [7]. The text encoder is built on Transformer [30] and generates text embeddings from the prompt. Specifically, "prompt" here refers to a sequence of words (tokens), such as "a neuropathological image of grey matter".

Training. CLIP jointly trains $F(\cdot)$ and $G(\cdot)$ to optimize the similarity score (e.g., symmetric cross-entropy loss [33]) between the visual and textual embeddings for each batch. The input consists of an image and its corresponding prompt (e.g., "this is a photo of grey matter"). They formulate the learning objective as a contrastive loss to align the visual and textual embedding spaces. Given a batch of image-prompt pairs, CLIP maximizes the similarity score for positive pairs while minimizing it for negative pairs.

Contrastive loss function. In CLIP training, the agreement between a positive pair of vision and text embeddings is maximized by InfoNCE loss [24]:

$$l_i^{v\to t} = -\log \frac{\exp\left(\mathbf{sim}(v_i, t_i)/\gamma\right)}{\sum_{k=1}^{B} \exp\left(\mathbf{sim}(v_i, t_k)/\gamma\right)}, \tag{1}$$

where v, t are normalized vectors from vision and text encoders. (v_i, t_i) is a positive pair. **sim** is a function to measure the similarity of two vectors. B is the mini-batch size of the image-text pairs and γ is the learnable temperature. Therefore, there is only 1 positive text for the input i^{th} image but $N-1$ negative texts. $l_i^{v\to t}$ means the InfoNCE loss from an image i to texts, so as the text loss $l_i^{t\to v}$. Then the loss function for CLIP is as follows:

$$\mathcal{L}_{\text{CLIP}} = \frac{1}{2B} \sum_{i=1}^{B} \left(l_i^{v\to t} + l_i^{t\to v}\right). \tag{2}$$

Inference for classification tasks. For the inference process, an image x is transformed into a feature manifold $f \in \mathbb{R}^D$, where D is the feature dimension. Then, f is multiplied with a classifier weight matrix $\mathbf{W} \in \mathbb{R}^{D\times K}$, where K is the number of classes in the learning task. We get a K-dimensional logit after matrix multiplication. Then we apply softmax to convert this logit into a probability vector $p \in \mathbb{R}^K$ over the K classes. The whole process can be summarized as the following equation:

$$p_i = \frac{\exp(\mathbf{W_i}^T \times f)/\gamma}{\sum_{i=1}^{K} \exp(\mathbf{W_i}^T \times f)/\gamma}, \tag{3}$$

where γ is a temperature parameter learned by CLIP during training and $\mathbf{W_i}$ is the prototype weight vector for class-i.

Transferability. Compared to traditional classifier learning methods where the supervision is a fixed one-hot vector, language-vision pre-training utilizes an open-set visual concept that can be explored through the text encoder to learn a broader semantic space, which in turn makes the learned representations more diverse and adaptable to various downstream tasks (e.g., zero-shot learning). Therefore, we argue that such language-vision pre-training is helpful for medical problems where the domain is very distinctive from the natural images.

3.2 Residual Feature Refinement

In this subsection, we introduce Residual Feature Refinement (RFR) to learn the task-relevant context when we adapt CLIP to the downstream learning tasks. Unlike CoOp's prompt tuning, which may not address the domain shift issue between the natural and pathology images, we focus on fine-tuning the visual features f. However, simple fine-tuning of the entire network may fail in a new pathological task due to 1) the overfitting issue caused by a large number of parameters in CLIP and a shortage of training samples [13]; 2) catastrophic forgetting issue.

Table 1: Baseline. We fine-tune CLIP (ResNet-50) on PCam [31] with 1% training data. "No fine-tuning" is the best baseline due to the catastrophic forgetting issue.

#	Configuration	Accuracy
1	No fine-tuning	56.5%
2	Fine-tune 10 epochs	40.3%
3	Fine-tune 100 epochs	35.9%
4	With a lower learning rate on #3	45.6%

Importance of the CLIP's pre-trained knowledge. CLIP has shown its zero-shot ability and transferability to a wide range of downstream tasks. We argue that preserving pre-trained knowledge of CLIP is crucial to maintain its generalization and alleviate the overfitting issue when the new dataset is small. There are two reasons behind this. First, one recent NLP work [11] shows large pre-trained language models (PLMs) are well-calibrated on the masked language modeling tasks. However, such property cannot be retained through direct fine-tuning due to catastrophic forgetting. Specifically, the pre-trained features are only used for the initialization and will be distorted by the fine-tuning [11]. Inspired by this, we investigate whether this issue also appears in language-vision models. As shown in Table 1, "no fine-tuning" but zero-shot gets the best results compared to fine-tuning CLIP on 1% of training data in PCam [31]. We also attempt to lower the learning rate to 1×10^{-6}, but the results are still far from satisfactory. Therefore, we conclude direct fine-tuning may distort the pre-trained knowledge and result in a severe catastrophic forgetting issue. Besides the forgetting issue, overfitting is another common challenge when adapting a big model to a new task. To alleviate the above two issues, we propose a residual refinement architecture to preserve the pre-trained features and acquire rich semantic information for pathology image analysis.

Architecture overview. As shown in Fig. 2, given an image x, we get the visual feature f from the image encoder and compute the classifier weight $\mathbf{W}$ from the text encoder. Then we design trainable adaptive layers L to convert f into $L(f)$. L can be multiple layers of linear transformations in a "down-and-up" architecture. As shown in Fig. 2, we have four layers to transform the feature dimension as "1024-256-64-256-1024". This is one example: the layers are flexible with other dimensions. After this architecture, $L(f)$ can be of the same size of f and blended with f as follows:

$$f_i^* = (1 - \alpha_i)L(f_i) + \alpha_i f_i, \tag{4}$$

where $i = \{1, 2, 3, ..n\}$ and α is the self-adaptive residual ratio to balance the fine-tuned knowledge and the original CLIP's knowledge. Specifically, the first term adaptively learns knowledge from the new dataset, while the second term preserves the original knowledge from CLIP.

Then we adopt Eq. (3) with the new f^* to get the class probability vector and predict the category. During fine-tuning, the weights of the trainable layers are optimized through the symmetric contrastive loss used in CLIP [26]. The prompt can be "this is a photo of []", where "[]" is filled with the class name. We can incorporate pathological descriptions into the prompt to provide more domain information, e.g., "this is a **pathological breast** photo of []".

Hidden Representation Perturbation (HRP). Fine-tuning a pre-trained language-vision model on a small dataset can lead to potential issues such as overfitting or representation collapse [45]. To mitigate these issues, we propose Hidden Representation Perturbation (HRP) that is strategically integrated with the trainable layers. To balance the model's complexity and efficiency, we adopt a "four-layer" architecture with HRP, steering clear of deeper architectures to ensure a controlled number of trainable parameters. This precaution is crucial as an excess of trainable parameters may exacerbate overfitting, especially in scenarios characterized by limited resources, where only a sparse dataset is available for the new task [52].

To alleviate the overfitting issue in low-resource settings, we focus on adding noise during the training process, a widely-used NLP strategy for large language models to smoothen the optimization landscape. For example, NoisyTune [35] adds the perturbations to the pre-trained weights to improve the model's generalization. Inspired by these NLP strategies [19, 35, 45], we propose to add random noise to the feature f before trainable adapting layers fine-tuning it. The random noise can be Gaussian noise ε generated from $\mathcal{N}(0, \sigma^2)$. After this, the perturbed feature f is converted into $\hat{f} = f + \varepsilon$. Therefore, Eq. (4) is converted into:

$$f_i^* = (1 - \alpha_i) L(\hat{f}_i) + \alpha_i f_i. \tag{5}$$

Therefore, the residual feature will be perturbed with the Gaussian noise for the refinement, designed to alleviate the overfitting issue.

3.3 Self-adaptive residual ratio

In Section 3.2, we introduce residual feature refinement and the residual ratio α. α helps modify the degree of preserving the original knowledge for better performance. Empirically, α should be small if the domain gap between the source data and the downstream task is enormous, so more knowledge from the downstream task ($L(f)$) can be considered. However, if α is too small, the model can be overfitting as it will heavily rely on the fine-tuned features and disregard the pre-trained knowledge from CLIP. On the other hand, the model can also be underfitting if α is too large, as it may prevent learning from the fine-tuned features. Therefore, it is crucial to quantify the residual ratio α so the model can be well-trained instead of underfitting or overfitting.

Recent works such as Tip-Adapter [51] and CLIP-Adapter [9] involve feature refinement while they set such ratio as a fixed hyper-parameter that needs to be

tuned during the training process. We argue the residual ratio should be dynamic and self-adaptive during training instead of fixed, as the learning status changes along the training process. Hence, we propose a self-adaptive residual ratio in Path-CLIP.

DVC: Dual-view Vision Contrastive. Inspired by self-supervised learning (e.g., SimCLR [4]), the composition of data augmentations is relevant to define effective predictive tasks. Therefore, we propose Dual-view Vision Contrastive (DVC) to measure the model's learning status. Given an image x, we leverage two types of augmentations: "weak" and "strong" [28]. "weak" refers to the standard flip-and-shift augmentation strategy while "strong" refers to perturb the appearance (RandAugment [5] and CTAugment [2]). We propose to measure the distance between representations from both augmentations, e.g., cosine similarity has shown good performance in both visual contrastive [4] and textual contrastive [3] tasks.

With the input x, we will get $w(x)$ from the weak augmentation and $s(x)$ from the strong augmentation. We obtain $F(w(x))$ and $F(s(x))$ after the vision encoder F. We define the distance as τ:

$$\tau = \frac{1}{N}\sum_{i=1}^{N} cos(F(w(x_i)), F(s(x_i))), \tag{6}$$

where N is the number of samples in the training set, and cos is a cosine similarity. The model's learning status is relevant to its consistency across different perturbations. We traverse the entire training set and calculate its mean value to represent the model's performance.

Bound constraints. In addition to DVC which measures the predictive consistency of the model, we also select "accuracy" on the training set as a second metric to measure the learning status, denoted as T. Empirically, "accuracy" alone cannot fully tell if the model is well-fitted: 100% of accuracy may indicate overfitting, which is the biggest challenge when we adapt a large model to a downstream task where the training set is very small. However, "accuracy" is a good metric to show if the model is underfitting. Therefore, we propose to fuse DVC and "accuracy" to represent the model's status during training.

If we look at α and set it fixed, we can find the model is prone to overfit if α is too small as it will heavily rely on the fine-tuned knowledge; the model tends to underfit if α is too large as it prevents the model from learning new knowledge (directly use the original pre-trained knowledge). Therefore, we propose a dynamic way with lower and upper bounds to obtain α during training:

$$\alpha = \min(\max(\tau, T), 1 - \tau). \tag{7}$$

The bounds are associated with predictive consistency (DVC) and predictive accuracy on the training set. For example, if the model is overfitting, T will be large, and τ will be relatively small. Then α will be smaller in the next epoch to use less fine-tuned features. Therefore, our proposed bound constraints can effectively adjust α to alleviate the overfitting issue.

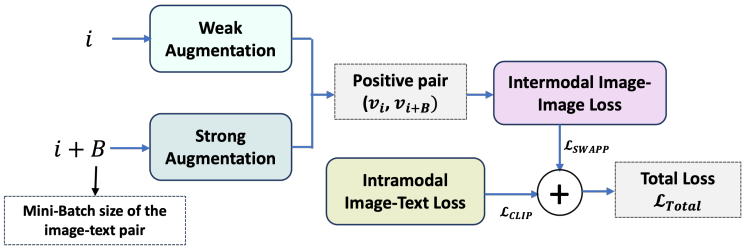


Fig. 3: Doublet Multimodal Contrastive Loss (DMCL) that utilizes the traditional image-text pair and the proposed image-image pair with weak/strong augmentation criteria.

3.4 Path-CLIP: End-to-end adaptation of CLIP to pathology tasks

To efficiently adapt CLIP to pathology image tasks, we propose RFR with a self-adaptive residual ratio to balance the original and fine-tuned knowledge based on the learning status. To alleviate the overfitting issue, we incorporate HRP and design bound constraints for the residual ratio. Besides these, to distill more visual representations from the new image domain, we add a contrastive loss to the loss function so that the trainable portion has a better classifying ability. We follow SimCLR [4] to define the Contrastive Loss (CL) for one image in the batch as:

$$l_i^{\mathrm{c}} = \begin{cases} -\log \frac{\exp(\mathbf{sim}(v_i, v_{i+B})/\gamma)}{\sum_{k\neq i}^{2B} \exp(\mathbf{sim}(v_i, v_k)/\gamma)} \text{ , if } \ i < B, \\ -\log \frac{\exp(\mathbf{sim}(v_i, v_{B-i})/\gamma)}{\sum_{k\neq i}^{2B} \exp(\mathbf{sim}(v_i, v_k)/\gamma)} \text{ , otherwise.} \end{cases} \tag{8}$$

Differing from SimCLR [4] and RepRec [38], as shown in Fig. 3, here we set image i as the weakly augmented image and image $i + B$ as the strongly augmented image instead of two randomly augmented images as the positive pair. We named this loss as $\mathcal{L}_{\mathbf{SWAPP}}$. **SWAPP** here refers to **S**trong-**W**eak **A**ugmented **P**ositive **P**air. Finally, the total loss for fine-tuning CLIP on our pathology datasets is as follows:

$$\begin{aligned} \mathcal{L}_{\text{total}} &= \mathcal{L}_{\text{CLIP}} + \mathcal{L}_{\text{SWAPP}}. \\ \mathcal{L}_{\text{CLIP}} &\rightarrow \frac{1}{2B} \sum_{i=1}^{B} \left(l_i^{v\rightarrow t} + l_i^{t\rightarrow v} \right), \\ \mathcal{L}_{\text{SWAPP}} &\rightarrow \frac{1}{2B} \sum_{i=1}^{2B} l_i^{\mathrm{c}} \end{aligned} \tag{9}$$

We define this loss as **Doublet Multimodal Contrastive Loss (DMCL)** because the items involve both intra-contrastive (image and text) and inter-contrastive (image and image). The algorithm can be found in the Appendix.

Table 2: Quantitative comparison on PCam. Precision, Recall, and F1-score refer to the macro-averaged values from all classes

Learning Manner	Labeled Ratio	Algorithm	Accuracy	Precision	Recall	F1-score	AUC
Supervised	100%	-	92.8 ± 0.30	92.9 ± 0.20	92.6 ± 0.21	92.8 ± 0.20	0.95 ± 0.01
Supervised	0.5%	-	45.1 ± 2.27	42.0 ± 2.14	33.6 ± 3.30	37.3 ± 2.86	0.51 ± 0.01
Self-supervised	0.5%	SimCLR [4]	60.4 ± 1.45	63.3 ± 1.90	58.5 ± 2.01	60.8 ± 1.20	0.61 ± 0.04
Semi-supervised	0.5%	Pseudo-Label [20]	55.7 ± 2.05	59.2 ± 2.23	54.5 ± 1.97	56.8 ± 1.55	0.58 ± 0.03
		FixMatch [28]	73.2 ± 0.76	77.5 ± 0.50	73.7 ± 0.25	75.6 ± 0.37	0.84 ± 0.04
		Dash [37]	71.0 ± 0.98	75.3 ± 0.78	70.2 ± 0.45	72.7 ± 0.50	0.82 ± 0.02
		FlexMatch [49]	74.0 ± 0.80	78.1 ± 1.15	73.0 ± 0.90	75.5 ± 0.84	0.85 ± 0.03
Multimodal CLIP	-	Zero-shot [26]	56.5 ± 0.00	57.4 ± 0.00	50.3 ± 0.00	53.7 ± 0.00	0.60 ± 0.00
	0.5%	CoOp [54]	63.6 ± 0.25	63.9 ± 0.42	62.5 ± 0.35	63.0 ± 0.29	0.67 ± 0.02
		CLIP-Adapter [9]	72.3 ± 1.02	77.2 ± 0.95	63.2 ± 0.56	69.4 ± 0.84	0.81 ± 0.02
		TIP-Adapter [51]	73.9 ± 0.50	70.5 ± 0.85	78.0 ± 0.49	74.0 ± 0.95	0.84 ± 0.02
		Path-CLIP	**81.9 ± 0.78**	**79.4 ± 0.35**	**85.0 ± 0.80**	**82.1 ± 0.95**	**0.89 ± 0.02**
Supervised	1.5%	-	62.8 ± 1.31	69.5 ± 0.67	54.1 ± 2.27	60.8 ± 1.68	0.63 ± 0.02
Self-supervised	1.5%	SimCLR [4]	64.9 ± 1.02	65.2 ± 1.45	60.1 ± 1.70	62.5 ± 1.05	0.64 ± 0.03
Semi-supervised	1.5%	Pseudo-Label [20]	67.9 ± 1.85	70.5 ± 1.90	62.2 ± 1.76	66.1 ± 1.42	0.67 ± 0.04
		FixMatch [28]	83.9 ± 1.29	85.1 ± 0.85	84.1 ± 0.95	84.6 ± 0.90	0.87 ± 0.01
		Dash [37]	82.8 ± 1.09	83.0 ± 0.55	81.2 ± 0.60	82.1 ± 0.78	0.85 ± 0.03
		FlexMatch [49]	84.1 ± 1.02	84.9 ± 0.75	84.4 ± 0.80	84.6 ± 0.75	0.86 ± 0.04
Multimodal CLIP	1.5%	CoOp [54]	61.5 ± 0.75	60.5 ± 1.03	65.6 ± 0.90	63.1 ± 0.85	0.65 ± 0.02
		CLIP-Adapter [9]	74.1 ± 0.88	79.1 ± 0.34	65.4 ± 0.89	71.6 ± 1.02	0.82 ± 0.01
		TIP-Adapter [51]	75.2 ± 0.94	72.3 ± 0.69	79.4 ± 0.86	75.7 ± 0.73	0.83 ± 0.02
		Path-CLIP	**83.1 ± 0.67**	**82.8 ± 0.95**	**82.0 ± 0.75**	**82.4 ± 0.46**	**0.91 ± 0.02**

4 Experiments

Datasets. We select two classification datasets (PCam [31] and MHIST [34]) and a gigapixel grey/white matter segmentation dataset to evaluate Path-CLIP. The details of these datasets can be found in Appendix.

Experimental setup. For a fair comparison, we use ResNet-50 [12] as the backbone in the vision encoder $f(\cdot)$ and BERT [6] as the textual encoder. We follow CLIP [26] to use gradient scaling to facilitate mixed-precision training. We set the learning rate as 0.0001. The batch size is 32. We use Adam [16] as the optimizer. To compare the computational complexity, we conduct all experiments on a single GPU (Nvidia RTX 2080Ti).

Baselines. We first implement traditional supervised learning (SL) methods with the entire dataset to explore the upper-bound performance for each task. Then we reduce the data usage and investigate the performance change of these SL methods. Then we compare Path-CLIP with original CLIP [26] and three recent CLIP-based fine-tuning methods: CoOp [54], Tip-Adapter [51] and CLIP-Adapter [9]. We compare them under the limited data scenarios to test their ability to do downstream tasks when the new dataset is small. Besides that, we also incorporate recent self-supervised learning (SimCLR [4]) and semi-supervised learning (Pseudo-Label [20], FixMatch [28], Dash [37], and FlexMatch [49]). Semi-supervised training requires a large amount of unlabeled data, which involves data collection efforts [41–43].

4.1 Main results

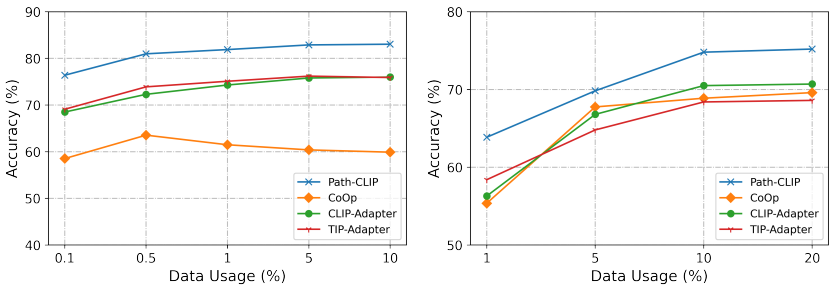
Classification task: PCam [31] and MHIST [34]. The main results for two classification benchmarks are summarized in Table 2 and Table 3. First, CLIP

Table 3: Quantitative comparison on MHIST. Precision, Recall, and F1-score refer to the macro-averaged values from all classes

Learning Manner	Labeled Ratio	Algorithm	Accuracy	Precision	Recall	F1-score	AUC
Supervised	100%	-	86.0 ± 0.36	84.8 ± 0.61	83.5 ± 0.55	84.1 ± 0.58	0.89 ± 0.01
Self-supervised	10%	SimCLR [4]	63.2 ± 0.95	63.8 ± 1.20	62.2 ± 1.70	63.0 ± 0.85	0.66 ± 0.04
Semi-supervised	10%	Pseudo-Label [20]	65.4 ± 1.04	66.5 ± 0.98	65.9 ± 0.85	66.2 ± 1.42	0.75 ± 0.03
		FixMatch [28]	70.5 ± 1.55	68.8 ± 2.76	69.5 ± 2.97	69.1 ± 2.86	0.79 ± 0.02
		Dash [37]	70.2 ± 1.21	70.0 ± 0.85	71.3 ± 0.69	70.6 ± 1.48	0.81 ± 0.04
		FlexMatch [49]	73.1 ± 0.84	74.6 ± 1.05	75.1 ± 0.90	74.8 ± 0.89	0.82 ± 0.03
Multimodal CLIP	10%	Zero-shot [26]	36.9 ± 0.00	36.9 ± 0.00	100 ± 0.00	53.9 ± 0.00	0.50 ± 0.00
		CoOp [54]	68.9 ± 0.40	56.0 ± 0.65	72.8 ± 0.53	63.3 ± 0.86	0.76 ± 0.01
		CLIP-Adapter [9]	65.9 ± 1.45	52.2 ± 1.23	87.2 ± 0.85	65.3 ± 0.95	0.77 ± 0.02
		TIP-Adapter [51]	63.1 ± 0.78	49.9 ± 0.90	94.7 ± 1.02	65.4 ± 0.56	0.78 ± 0.03
		Path-CLIP	**74.8 ± 1.95**	**63.3 ± 1.40**	**75.6 ± 0.98**	**68.9 ± 1.50**	**0.84 ± 0.02**

shows its promising zero-shot ability: when we directly test CLIP on PCam [31], it can get 56.5% accuracy. Then, we compare Path-CLIP with other CLIP-based fine-tuning methods. In PCam [31], we present 0.5% and 1.5% data usage in the training set to fine-tune CLIP, respectively. We find all of these fine-tuning algorithms [9, 51, 54] can improve the performance of the original CLIP with the new data from the downstream task presented. Path-CLIP can outperform them in both settings. Specifically, Path-CLIP outperforms TIP-Adapter [51] with 8% and 7.9% accuracy when only 0.5% and 1.5% data are used, respectively. Path-CLIP also shows similar improvement in MHIST [34] (Table 3). We also implemented several semi-supervised learning where the rest of the data in the training set are used as an unlabeled set. Although Path-CLIP does not use unlabeled data, it can achieve competitive results with semi-supervised learning.

Besides these ratios of data usage, we also visualize the results by varying the ratio to show the performance of each fine-tuning algorithm. As shown in Fig. 4, CoOp [54] may not improve PCam when more data are used to fine-tune the prompt. CoOp freezes the encoders and only fine-tunes the language prompt part, while other algorithms also fine-tune the vision part. We conclude that visual fine-tuning is more critical compared to prompt tuning.



(a) PCam (b) MHIST

Fig. 4: Comparison of different fine-tuning methods by varying data usage.

Table 4: Performance comparison on GM/WM segmentation.

Data Usage	Algorithm	GM_IoU	GM_DICE	WM_IoU	WM_DICE
100% (SL)	FCN [1]	71.72	83.21	49.09	66.08
	U-Net [25]	91.52	95.17	82.02	88.15
0.1% (SL)	FCN [1]	48.25	25.01	24.54	30.02
	U-Net [25]	46.13	50.10	5.75	12.23
0.1% (CLIP)	CoOp [54]	62.23	65.70	39.92	48.65
	CLIP-Adapter [9]	67.80	74.32	43.25	54.61
	TIP-Adapter [51]	69.92	75.50	46.23	56.60
	Path-CLIP	**78.05**	**83.70**	**54.50**	**63.23**
1% (SL)	FCN [1]	51.34	30.20	31.02	35.12
	U-Net [25]	50.24	53.23	15.43	25.58
1% (CLIP)	CoOp [54]	63.45	64.12	37.85	45.34
	CLIP-Adapter [9]	69.99	76.54	47.95	58.34
	TIP-Adapter [51]	72.12	77.30	55.65	63.60
	Path-CLIP	**80.21**	**88.70**	**60.50**	**70.34**

GM/WM segmentation [17]. Besides two classification tasks, we follow [17] to generalize our method to a weakly-supervised segmentation problem. We select IoU and DICE scores as the performance metrics. The results are summarized in Table 4. We select FCN [1] and U-Net [25] as two SL baselines: we first train them on all data (100%) in a supervised manner, then we reduce the dataset to 0.1% and 1% setting and train the model again. We can see both SL algorithms face performance degradation when the training data is scarce. They require an extensive and well-curated labeled dataset to get satisfactory results. We then focus on fine-tuning CLIP algorithms. Table 4 shows Path-CLIP can achieve superior results than other fine-tuning techniques.

4.2 Ablation studies

Importance of alleviating overfitting. Since the new task may have a very limited training set, it is crucial to alleviate the potential overfitting issue. Besides the RFR with the self-adaptive ratio, we also propose HRP, DVC, and DMCL to cope with the overfitting issue. Table 5 summarizes the contribution of each proposed module: HRP can improve with +2.2% accuracy and DVC can improve with +4.3%. DMCL inspired by self-supervised learning can also aid in great improvements. We conclude that contrastive learning, perturbations on data and hidden representations are helpful to the overfitting issue.

Computational efficiency. When we range the data usage from 0.1% to 1%, Path-CLIP can obtain +5.5% gain in accuracy while only introducing 1 minute of additional training time. However, CoOp [54] has the overfitting issue and obtains a lower score while using 53 minutes, almost five times compared to the time used by RFR. The situation remains similar under the 10% of data usage. We conclude our proposed RFR can get over 19% improvement on CLIP by only

Table 5: Ablation study on PCam using 0.5% data for fine-tuning.

#	RFC	HRP	DVC	DMCL	Accuracy	F1-score	AUC
1	✗	✗	✗	✗	56.5	53.7	0.60
2	✓	✗	✗	✗	72.0	70.5	0.69
3	✓	✓	✗	✗	74.2	75.5	0.78
4	✓	✓	✓	✗	78.5	79.3	0.84
5	✓	✓	✓	✓	81.9	82.1	0.89

using 10 minutes for the fine-tuning on a single GPU, which is promising for digital pathology research. The details can be found in Table 8 in the Appendix.

Generalization to pathology foundation model: PLIP [14]. The results of using PLIP's weights are in Appendix 3.6: PLIP is pre-trained on a large-scale pathology dataset and achieves strong zero-shot performance. With our method, we can further boost the performance and outperform other tuning methods.

5 Discussion

In this work, we explore the generalization of CLIP in pathology image analysis. To alleviate the overfitting and forgetting issues when adapting CLIP to small downstream datasets, we propose self-adaptive RFR with HRP. We propose Path-CLIP with DMCL as an end-to-end pipeline for CLIP adaptation in pathology tasks. The results reveal the superiority and effectiveness of Path-CLIP compared to the existing fine-tuning methods. In addition, the less computational time improves the efficiency and accessibility of Path-CLIP. Hence, we conclude Path-CLIP is more robust and has the potential to bridge the domain shift between the pre-trained natural images and pathology images in an efficient and cost-effective manner.

Acknowledgments

This work was supported by the Noyce Initiative UC Partnerships in Computational Transformation Grant and Child Family Endowed Professorship. Resources for this study were funded in part by grants from the National Institute on Aging of the National Institutes of Health under Award Numbers R01AG062517, P30AG072972, and R01AG056519.

References

1. Bándi, P., van de Loo, R., Intezar, M., Geijs, D., Ciompi, F., van Ginneken, B., van der Laak, J., Litjens, G.: Comparison of different methods for tissue segmentation in histopathological whole-slide images. In: Proc. 2017 IEEE Int. Symp. Biomed. Imaging. pp. 591–595 (2017)
2. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.A.: MixMatch: A holistic approach to semi-supervised learning. In: NeurIPS. vol. 32 (2019)

3. Chen, F., Zhang, H., Li, Z., Dou, J., Mo, S., Chen, H., Zhang, Y., Ahmed, U., Zhu, C., Savvides, M.: Unitail: Detecting, reading, and matching in retail scene. In: ECCV. pp. 705–722. Springer (2022)
4. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: ICML. pp. 1597–1607 (2020)
5. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: Practical automated data augmentation with a reduced search space. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshop. pp. 702–703 (2020)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
7. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
8. Gal, R., Patashnik, O., Maron, H., Bermano, A.H., Chechik, G., Cohen-Or, D.: Stylegan-nada: Clip-guided domain adaptation of image generators. ACM Transactions on Graphics (TOG) **41**(4), 1–13 (2022)
9. Gao, P., Geng, S., Zhang, R., Ma, T., Fang, R., Zhang, Y., Li, H., Qiao, Y.: Clip-adapter: Better vision-language models with feature adapters. arXiv preprint arXiv:2110.04544 (2021)
10. Ge, C., Huang, R., Xie, M., Lai, Z., Song, S., Li, S., Huang, G.: Domain adaptation via prompt learning. arXiv preprint arXiv:2202.06687 (2022)
11. He, G., Chen, J., Zhu, J.: Preserving pre-trained features helps calibrate fine-tuned language models. In: ICLR (2023)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
13. Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., Gelly, S.: Parameter-efficient transfer learning for nlp. In: ICML. pp. 2790–2799. PMLR (2019)
14. Huang, Z., other: A visual–language foundation model for pathology image analysis using medical twitter. Nature medicine (2023)
15. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: ICML. pp. 4904–4916 (2021)
16. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (Poster) (2015)
17. Lai, Z., Oliveira, L.C., Guo, R., Xu, W., Hu, Z., Mifflin, K., Decarli, C., Cheung, S.C., Chuah, C.N., Dugger, B.N.: Brainsec: Automated brain tissue segmentation pipeline for scalable neuropathological analysis. IEEE Access **10**, 49064–49079 (2022)
18. Lai, Z., Wang, C., Oliveira, L.C., Dugger, B.N., Cheung, S.C., Chuah, C.N.: Joint semi-supervised and active learning for segmentation of gigapixel pathology images with cost-effective labeling. In: ICCV Workshop. pp. 591–600 (2021)
19. Lam, S.Y., Zeng, Q., Zhang, K., You, C., Voigt, R.: Large language models are partially primed in pronoun interpretation. arXiv preprint arXiv:2305.16917 (2023)
20. Lee, D.H.: Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: Workshop on challenges in representation learning (ICML) (2013)
21. Liu, M., Hu, L., Tang, Y., Wang, C., He, Y., Zeng, C., Lin, K., He, Z., Huo, W.: A deep learning method for breast cancer classification in the pathology images. IEEE J. Biomed. Health. Inf. **26**(10), 5025–5032 (2022)

22. Lu, Y., Liu, J., Zhang, Y., Liu, Y., Tian, X.: Prompt distribution learning. In: CVPR. pp. 5206–5215 (2022)
23. Lyu, W., Dong, X., Wong, R., Zheng, S., Abell-Hart, K., Wang, F., Chen, C.: A multimodal transformer: Fusing clinical notes with structured ehr data for interpretable in-hospital mortality prediction. In: AMIA Annual Symposium Proceedings. vol. 2022, p. 719. American Medical Informatics Association (2022)
24. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
25. Oskal, K.R., Risdal, M., Janssen, E.A., Undersrud, E.S., Gulsrud, T.O.: A U-net based approach to epidermal tissue segmentation in whole slide histopathological images. SN Appl. Sci. **1**(7), 1–12 (2019)
26. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML. pp. 8748–8763 (2021)
27. Rasheed, H., khattak, M.U., Maaz, M., Khan, S., Khan, F.S.: Finetuned clip models are efficient video learners. In: The IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
28. Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C.A., Cubuk, E.D., Kurakin, A., Li, C.L.: FixMatch: Simplifying semi-supervised learning with consistency and confidence. In: NeurIPS. vol. 33, pp. 596–608 (2020)
29. Tian, Y., Newsam, S., Boakye, K.: Fashion image retrieval with text feedback by additive attention compositional learning. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1011–1021 (2023)
30. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. NeurIPS **30** (2017)
31. Veeling, B.S., Linmans, J., Winkens, J., Cohen, T., Welling, M.: Rotation equivariant cnns for digital pathology. In: MICCAI. pp. 210–218. Springer (2018)
32. Wang, X., Yang, S., Zhang, J., Wang, M., Zhang, J., Huang, J., Yang, W., Han, X.: Transpath: Transformer-based self-supervised learning for histopathological image classification. In: MICCAI. pp. 186–195. Springer (2021)
33. Wang, Y., Ma, X., Chen, Z., Luo, Y., Yi, J., Bailey, J.: Symmetric cross entropy for robust learning with noisy labels. In: ICCV (2019)
34. Wei, J., Suriawinata, A., Ren, B., Liu, X., Lisovsky, M., Vaickus, L., Brown, C., Baker, M., Tomita, N., Torresani, L., et al.: A petri dish for histopathology image analysis. In: Int. Conf. Arti. Intell. in Medi. pp. 11–24. Springer (2021)
35. Wu, C., Wu, F., Qi, T., Huang, Y.: Noisytune: A little noise can help you finetune pretrained language models better. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 680–685 (2022)
36. Xu, P., Zhu, X., Clifton, D.A.: Multimodal learning with transformers: A survey. arXiv preprint arXiv:2206.06488 (2022)
37. Xu, Y., Shang, L., Ye, J., Qian, Q., Li, Y.F., Sun, B., Li, H., Jin, R.: Dash: Semi-supervised learning with dynamic thresholding. In: International Conference on Machine Learning. pp. 11525–11536. PMLR (2021)
38. Yan, X., Naushad, J., Sun, S., Han, K., Tang, H., Kong, D., Ma, H., You, C., Xie, X.: Representation recovering for self-supervised pre-training on medical images. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 2685–2695 (2023)
39. Yang, B., Liu, F., Li, Z., Yin, Q., You, C., Yin, B., Zou, Y.: Multimodal prompt learning for product title generation with extremely limited labels. arXiv preprint arXiv:2307.01969 (2023)

40. Yang, J., Chen, H., Liang, Y., Huang, J., He, L., Yao, J.: Concl: Concept contrastive learning for dense prediction pre-training in pathology images. In: ECCV. pp. 523–539. Springer (2022)
41. You, C., Dai, W., Min, Y., Liu, F., Zhang, X., Feng, C., Clifton, D.A., Zhou, S.K., Staib, L.H., Duncan, J.S.: Rethinking semi-supervised medical image segmentation: A variance-reduction perspective. In: NeurIPS (2023)
42. You, C., Dai, W., Min, Y., Staib, L., Duncan, J.S.: Bootstrapping semi-supervised medical image segmentation with anatomical-aware contrastive distillation. In: International Conference on Information Processing in Medical Imaging. pp. 641–653. Springer (2023)
43. You, C., Dai, W., Min, Y., Staib, L., Sekhon, J., Duncan, J.S.: Action++: Improving semi-supervised medical image segmentation with adaptive anatomical contrast. arXiv preprint arXiv:2304.02689 (2023)
44. Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., Wu, Y.: Coca: Contrastive captioners are image-text foundation models. arXiv preprint arXiv:2205.01917 (2022)
45. Yuan, H., Yuan, Z., Tan, C., Huang, F., Huang, S.: Hype: Better pre-trained language model fine-tuning with hidden representation perturbation. arXiv preprint arXiv:2212.08853 (2022)
46. Yuan, L., Chen, D., Chen, Y.L., Codella, N., Dai, X., Gao, J., Hu, H., Huang, X., Li, B., Li, C., et al.: Florence: A new foundation model for computer vision. arXiv preprint arXiv:2111.11432 (2021)
47. Zhai, X., Kolesnikov, A., Houlsby, N., Beyer, L.: Scaling vision transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12104–12113 (June 2022)
48. Zhai, X., Wang, X., Mustafa, B., Steiner, A., Keysers, D., Kolesnikov, A., Beyer, L.: Lit: Zero-shot transfer with locked-image text tuning. In: CVPR. pp. 18123–18133 (2022)
49. Zhang, B., Wang, Y., Hou, W., Wu, H., Wang, J., Okumura, M., Shinozaki, T.: Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. NeurIPS **34** (2021)
50. Zhang, R., Guo, Z., Zhang, W., Li, K., Miao, X., Cui, B., Qiao, Y., Gao, P., Li, H.: Pointclip: Point cloud understanding by clip. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8552–8562 (2022)
51. Zhang, R., Zhang, W., Fang, R., Gao, P., Li, K., Dai, J., Qiao, Y., Li, H.: Tip-adapter: Training-free adaption of clip for few-shot classification. In: ECCV. pp. 493–510. Springer (2022)
52. Zhang, T., Wu, F., Katiyar, A., Weinberger, K.Q., Artzi, Y.: Revisiting few-sample bert fine-tuning. In: ICLR (2021)
53. Zhang, W., Zhu, L., Hallinan, J., Zhang, S., Makmur, A., Cai, Q., Ooi, B.C.: Boostmis: Boosting medical image semi-supervised learning with adaptive pseudo labeling and informative active annotation. In: CVPR. pp. 20666–20676 (2022)
54. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Learning to prompt for vision-language models. Int. J. Comput. Vision **130**(9), 2337–2348 (2022)