FLIER: Few-shot Language Image Models Embedded with Latent Representations

Zhinuo Zhou[1], **Peng Zhou**[2], **Xiaoyong Pan**[1*]

[1]Shanghai Jiao Tong University
[2]Second Institute of Oceanography, Ministry of Natural Resources

Abstract

As the boosting development of large vision-language models like Contrastive Language-Image Pre-training (CLIP), many CLIP-like methods have shown impressive abilities on visual recognition, especially in low-data regimes scenes. However, we have noticed that most of these methods are limited to introducing new modifications on text and image encoder. Recently, latent diffusion models (LDMs) have shown good ability on image generation. The potent capabilities of LDMs direct our focus towards the latent representations sampled by UNet. Inspired by the conjecture in CoOp that learned prompts encode meanings beyond the existing vocabulary, we assume that, for deep models, the latent representations are concise and accurate understanding of images, in which high-frequency, imperceptible details are abstracted away. In this paper, we propose a Few-shot Language Image model Embedded with latent Representations (FLIER) for image recognition by introducing a latent encoder jointly trained with CLIP's image encoder, it incorporates pre-trained vision-language knowledge of CLIP and the latent representations from Stable Diffusion. We first generate images and corresponding latent representations via Stable Diffusion with the textual inputs from GPT-3. With latent representations as "models-understandable pixels", we introduce a flexible convolutional neural network with two convolutional layers to be the latent encoder, which is simpler than most encoders in vision-language models. The latent encoder is jointly trained with CLIP's image encoder, transferring pre-trained knowledge to downstream tasks better. Experiments and extensive ablation studies on various visual classification tasks demonstrate that FLIER performs state-of-the-art on 11 datasets for most few-shot classification.

Introduction

Numerous efficient models for addressing visual tasks (such as ResNet(He et al. 2016), ViT(Dosovitskiy et al. 2020), CLIP(Radford et al. 2021), BEiT(Bao et al. 2021), MAE(He et al. 2022), CAE(Chen et al. 2024)), have achieved continuous improvements in model performance across various datasets. However, it is observed that in real-world application scenarios, the quantity or quality of available data often falls significantly below the standards of existing comprehensive datasets. CLIP, as a representative model of visual-language fusion, holds significant importance in achieving

strong few-shot generalization performance across various computer vision tasks, particularly for zero-shot image classification. Since the introduction of CLIP, numerous models have emerged aiming to optimize its performance, including CoOp(Zhou et al. 2022b), Clip-Adapter(Gao et al. 2023), Tip-Adapter(Zhang et al. 2022), CaFo(Zhang et al. 2023). Of these models, some optimize the performance of CLIP through prompt learning. Others enhance the model's ability through adaptation and cache methods.

Recently, generative models have rapidly been developed. Initially, when generative models were first introduced, a series of models, like Generative Adversarial Networks (GAN)(Goodfellow et al. 2014) and variational autoencoders (VAE)(Kingma and Welling 2014), emerged prominently. GANs operate through a discriminator and a generator to accomplish the process of image generation, generating high resolution images with good quality. Different from GANs, VAE generates high resolution images efficiently through stochastic variational inference and learning. After GANs and VAE, diffusion models, like DALL-E(Ramesh et al. 2021) and Stable Diffusion 2(Rombach et al. 2022), have gained considerable attention due to their superior generation results. By introducing noise to degrade images, diffusion models employ a learned reverse process, effectively denoising to reconstruct the original images based on stochastic processes. In the generation process of Stable Diffusion 2 (SD2), it utilizes CLIP to encode the condition information and employs a trained sampler to sample Gaussian noise, obtaining image embeddings in the latent space. Finally, embeddings are decoded through a decoder to generate images. We notice that when image embeddings are transformed into images, they may be unclear for humans to recognize. However, the decoder can decode these embeddings and produce accurate images. This observation leads us to pose a question: "Considering these embeddings are created by models themselves, are they potentially more understandable for models as latent representations?". With this in mind, we integrate the image embeddings into CLIP (the CLIP image encoder to be more specific) via simple Convolutional neural networks (CNN)(Krizhevsky, Sutskever, and Hinton 2012), aiming to enhance the visual-language model.

In this paper, we propose FLIER, a novel visual-language model integrated with generative modules, for few-shot im-

---
*Corresponding authors.

age recognition. In Figure 1, the pipeline of FLIER starts from Prompting. To generate input textual prompts for SD2 corresponding to given category names, we employ GPT-3 to get rich additional information relevant to each category. Then we utilize SD2 to generate images from texts in Prompting. In the process of conditioned generation of SD2, we save the latent embeddings along with their corresponding generated images by the decoder, which not only enlarges the few-shot training data, but also gets the latent representations. Finally for joint training, we divide the training into two parts. In the first part, the image encoder of CLIP is trained by connecting linear probe on the training images from the original dataset. In the second part, the input of the image encoder is the generated images. Meanwhile, the latent encoder, a simple CNN with two convolutional layers, takes the latent representations as input. Finally, the models are trained jointly, and the loss is calculated by the weighted losses from the two models with a latent factor α. This approach allows CLIP to be thoroughly trained on the original dataset, while incorporating latent representations into the training process in a straightforward way.

Our main contributions are summarized as follows:

1. We propose FLIER, a new visual-language model based on CLIP for few-shot classification tasks, to integrate prior vision-language knowledge with latent representations in diffusion modules for better representation learning.

2. Different from using frozen pre-trained weights, FLIER trains the image encoder of CLIP with the latent encoder jointly, which guarantees a new understanding of images injected into CLIP, improving FLIER's ability on transferring prior pre-trained knowledge and latent representations to downstream classification tasks with a state-of-the-art (SOTA) on ImageNet(Deng et al. 2009).

3. We perform extensive ablation studies of FLIER on ImageNet to show the effectiveness of individual modules and evaluate FLIER on 11 benchmark datasets for few-shot classification, where FLIER achieves SOTA in most experiments without using additional annotated data.

Related Work

Vision-Language Models

With the continuous development of language and vision models, there has been an increasing number of methods for exploring the interaction between vision and language. After the effectiveness of attention mechanisms was proved, in vision-language models, attention-based approaches have demonstrated excellent performance such as BAN(Kim, Jun, and Zhang 2018), Intra-Inter(Gao et al. 2019) and MCAN(Yu et al. 2019). Since BERT(Devlin et al. 2018) showed impressive performance, several subsequent works(Lu et al. 2019; Tan and Bansal 2019) based on BERT further propelled the development of visual language models.

Recent vision-language models(Fürst et al. 2022; Jia et al. 2021; Li et al. 2021), represented by CLIP(Radford et al. 2021), connect visual and language knowledge by learning image and text encoders jointly. Compared to previous models, these models leverage contrastive representa-tion learning(Chen et al. 2020; He et al. 2020; Henaff 2020) to fully utilize the general knowledge learned from large-scale training datasets. The remarkable success of CLIP in zero-shot image recognition inspires related research on CLIP-like models(Dong et al. 2022; Zhou et al. 2022b,a; Gao et al. 2023; Zhang et al. 2022). Some CLIP-based methods like CoOp(Zhou et al. 2022b) introduce prompt design to improve models. Adaption(Gao et al. 2023; Zhang et al. 2022) and fine-tuning(Dong et al. 2022) on CLIP also achieve outstanding performance. As the development of image generation, diffusion models attract great attention. Based on both vision-language models and generative models, CaFo(Zhang et al. 2023) incorporates the knowledge of CLIP(Radford et al. 2021), DINO(Caron et al. 2021), DALL-E(Ramesh et al. 2021) and GPT-3(Brown et al. 2020) for improving the few-shot classification performance. Different from CaFo, we introduce a new idea about "models-understandable pixels" and our FLIER jointly trains a latent encoder for latent representations with CLIP's image encoder. Specifically, the latent encoder is a simple backbone consisting of two convolutional layers.

Diffusion Models

Recently, in the field of density estimation (Kingma et al. 2021) and sample quality (Dhariwal and Nichol 2021), compared to previous methods, diffusion probabilistic models (DM) (Sohl-Dickstein et al. 2015) demonstrate excellent performance. These models use UNet as the backbone for generating images to fit inductive deviations naturally(Dhariwal and Nichol 2021; Ho, Jain, and Abbeel 2020; Ronneberger, Fischer, and Brox 2015; Song et al. 2020). However, due to the complicated model structure and operating pixel space, evaluating and training these models has shortcomings that they infer slowly and cost a lot when training. To address these drawbacks, SD2(Rombach et al. 2022) introduces cross-attention layers into the architecture and changes the operating space to lower dimensional compressed latent space in a two-stage image synthesis way. It speeds up the inference with almost no reduction in synthesis quality. Inspired by the exciting results of SD2 and the detachable architecture, we isolated the latent representations generated by UNet during the inference for subsequent joint training with CLIP.

Few-Shot Learning

Few-shot learning (FSL) aims to enable models to learn and generalize to classes with a limited number of annotated samples(Wang et al. 2020). In recent years, FSL typically focuses on three perspectives: data, model, and algorithm, to improve model's performance. At the data level, FSL often employs prior knowledge to augment and enhance datasets(Douze et al. 2018; Pfister, Charles, and Zisserman 2014). In the meantime, similar datasets can provide relative prior knowledge(Gao et al. 2018; Tsai and Salakhutdinov 2017). At the model level, FSL focuses on narrowing the hypothesis space and searching optimal parameters for the model with prior knowledge. Some model-based optimization methods(Benaim and Wolf 2018; Liu et al. 2018; Cai et al. 2018; Ramalho and Garnelo 2019) primarily enhance

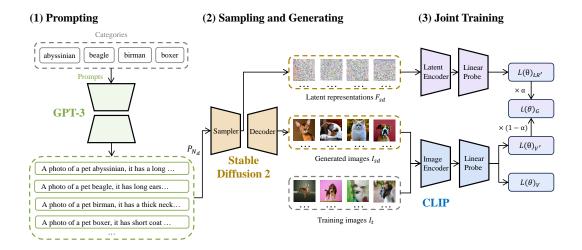**(1) Prompting**   **(2) Sampling and Generating**   **(3) Joint Training**

Figure 1: Overview of FLIER. First, we use GPT-3 to generate prompts for each class. Then, we input the prompts into SD2 and obtain latent representations and generated images. Finally, the latent encoder and CLIP's image encoder are jointly trained with latent representations, generated images $I_{N_d,K'}$ and training images $I_{N_d,K}$.

the structure and design of models through four aspects: multitask learning, embedding learning, learning with external memory, and generative modeling. Meanwhile, algorithmic optimization methods(Kozerawski and Turk 2018; Yu et al. 2018; Rusu et al. 2018; Ravi and Larochelle 2016) have traditionally concentrated on improving search strategies: how to refine existing parameters, refine meta-learning parameters, and learn optimizers.

With the emergence of visual-language pre-training models such as CLIP(Radford et al. 2021), optimizing vision-language models on few-shot datasets has attracted considerable attention. Some works(Zhou et al. 2022b; Zhang et al. 2023) do not require additional training of the model; instead, they achieve good performance on few-shot datasets by optimizing prompts or constructing key-value cache models. Other works(Dong et al. 2022; Zhang et al. 2022; Gao et al. 2023) achieve higher few-shot accuracy by introducing learnable parameters for training the model. Different from existing methods, we integrate latent representations with CLIP using simple customized networks.

FLIER Approach

An overview of our approach FLIER is shown in Figure 1. We first introduce the method of fine-tuning CLIP's image encoder. Then, we present the details of low-dimensional latent representation, which is the key component of FLIER. Finally, we elaborate the joint training details of FLIER.

Contrastive Language-Image Pre-training and Fine-tuning

Contrastive Language-Image Pre-training is a vision-language model consisting of two encoders for texts and images(Radford et al. 2021). With a backbone of a Transformer(Vaswani et al. 2017), the text encoder produces text

representations from encoded tokens of textual inputs. The backbone of image encoder is a ViT(Dosovitskiy et al. 2020) or ResNet(He et al. 2016), which produces the feature vectors for images. When training, CLIP utilizes cosine similarity to measure the matching degree of different text-image pairs. The optimization goal is to maximizes the cosine similarities of one image with the matched text, and minimizes that with unmatched texts for every pair. For the loss function, CLIP calculates the average cross entropy loss of two encoders. We let $I \in \mathbb{R}^{H \times W \times 3}$ be the image input, where H and W is the image's height and width. With the image encoder, an image feature $f \in \mathbb{R}^D$ is obtained from I, where D stands for the dimension of image's feature.

CLIP Fine-tuning   requires CLIP to use a linear probe evaluation protocol for downstream classification tasks. In this way, let $W \in \mathbb{R}^{D \times K}$ represents the weight of the linear layer connected to CLIP's image encoder, where K represents the number of categories. The image feature f is calculated through the linear layer to obtain a logit. The softmax function then converts the logits obtained by the linear layer into predicted probability $p \in \mathbb{R}^K$.

$$f = ImageEncoder(I), \quad logits_i = W_i^T f, \qquad (1)$$

$$p_i = \frac{exp(logits_i)/\gamma}{\sum_{j=1}^{N} exp(logits_j)/\gamma}. \qquad (2)$$

where γ represents the temperature of Softmax, W_i stands for the prototype weight vector for class i, and p_i denotes the probability of category i.

When fine-tuning CLIP (with linear probe), specifically the image encoder of CLIP, we apply strategies of the simultaneously updated exponential moving average (EMA) approach, the layer-wise learning rate (LLRD) strategy and initialing learning rate in fine-tuning. We finetune the CLIP on

11 datasets for various experiments for CLIP ViT-Base/16 with 224×224 input resolution on the whole ImageNet with configuration, including AdamW optimizer, cosine decay learning schedule, batch size 64, learning rate 0.0001, epochs 40, weight decay 0.05 and a latent factor 0.5 .

Prompting and Sampling Latent Representations

Prompting via GPT-3. Considering the experimental setting of few-shot classification, the number of generated images has to be larger than K when we conduct N-way and K-shot few-shot experiments. Thus, we use 10 effective prompts for each category. For different datasets d, let N_d represent the number of categories in d. For every N_d category, we try practical commands for GPT-3(Brown et al. 2020) such as "How do you describe a [CLASS]?", "What does a [CLASS] look like?" and "Tell me what is a [CLASS]". Let P_{N_d} stand for prompts generated by GPT-3.

$$P_{N_d} = \text{GPT-3(Commands)}. \tag{3}$$

Sampling Latent Representations via SD2. By text-to-image method of SD2, we adopt the prompts P_{N_d} as input. For model strategies, we load the checkpoint of SD2 with EMA for 512×512 images and use DPM(Lu et al. 2022) sampler for the inference. Under the setting of few-shot experiments, the maximum number required of each category N_d is 16. We denote the generated images of N_d as $I_{N_d,K'}$ and training images of N_d as $I_{N_d,K}$, where K and K' represent the K-shot setting in few-shot learning. In order to avoid images insufficiency, for each category, we set the batch size to be 2 and generate 20 images by 10 given prompts P_{N_d}, formulated as

$$I_{N_d,K'} = \text{Stable Diffusion 2}(P_{N_d}). \tag{4}$$

For each category, we obtain 20 generated images and their corresponding 20 latent representations. Also, we denotes $F_{N_d,n,K}$ as the latent representation of nth generated image in the category K of dataset N_d. In the few-shot experiments, to maintain the low-data rules, we keep K equal to K' all the time.

Joint Training.

We present a joint training framework by embedding low-dimensional latent representations into language-image encoder with a two-layer CNN for achieving better performance. It is difficult for CLIP's image encoder itself to fine-tune under the few-shot setting, since a small amount of data can hardly guarantee a large model to perform well in few-shot experiments. Different from appending additional learnable bottleneck of linear layers in CLIP-Adapter(Gao et al. 2023), we train a latent encoder with CLIP's image encoder jointly. Let Ψ_l represent the latent encoder, Ψ_v represent the image encoder of CLIP, W_l represent the linear probe connected with Ψ_l and W_v represent the linear probe connected with Ψ_v. We obtain the embeddings by two encoders and calculate the logits by linear probe:

$$f_I = \Psi_v(I_{N_d,K}), \ Logits_V = W_v^T f_I,$$
$$f_{I'} = \Psi_v(I_{N_d,K'}), \ Logits_{V'} = W_v^T f_{I'}, \tag{5}$$
$$f_{L'} = \Psi_l(F_{N_d,n,K'}), \ Logits_{L'} = W_l^T f_{L'}.$$

Then, we adopt Equation (2) to calculate $P_{V'} = \{p_{v',i}\}_{i=1}^K$, $P_V = \{p_{v,i}\}_{i=1}^K$ and $P_{LR'} = \{p_{lr',i}\}_{i=1}^K$, which denotes the category vector of the image encoder on the training data, the image encoder on the generative data and latent encoder with the latent representations.

Finally, for the loss function, we utilize label smooth cross entropy loss to prevent model over-fitting during fine-tuning. For joint training, we divide the training period into two parts. The first part is for training on the training dataset, optimizing the weights of the image encoder and its linear probe with the loss $L(\theta)_V$, which is calculated by Equation (6) and Equation (7). The second part is jointly training CLIP's image encoder with the latent encoder on the generated images, optimizing the weights of both encoders and their linear probe with the loss $L(\theta)_G$. The loss $L(\theta)_G$ consists of $L(\theta)_{V'}$ and $L(\theta)_{LG'}$, which are computed based on the image encoder and latent encoder, respectively. We utilize the latent factor α to measure the contribution of different losses to $L(\theta)_G$.

$$y_i = \begin{cases} \frac{\epsilon}{n} & i \neq target \\ 1 - \epsilon + \frac{\epsilon}{n} & i = target \end{cases}, \tag{6}$$

$$L(\theta)_S = -\frac{1}{N}\sum_i^N y_i log(p_{S,i}), \tag{7}$$

where S refers to LR', V' and V; n is the total number of N_d's categories; N is the total number of images in the dataset; $target$ is the ground-truth label; θ represents all learnable parameters; ϵ is the smoothing factor.

Experiments

Training Settings

Datasets. For few-shot classification and domain generalization evaluation, we conduct experiments for FLIER on 11 image classification datasets: ImageNet(Deng et al. 2009), OxfordPets(Parkhi et al. 2012), Caltech101(Fei-Fei, Fergus, and Perona 2004), SUN397(Xiao et al. 2010), Food101(Bossard, Guillaumin, and Van Gool 2014), DTD(Cimpoi et al. 2014), Flowers102(Nilsback and Zisserman 2008), EuroSAT(Helber et al. 2019), UCF101(Soomro, Zamir, and Shah 2012), StanfordCars(Krause et al. 2013) and FGVCAircraft(Maji et al. 2013). Specifically, we train our FLIER with 1, 2, 4, 8, 16 shots for few-shot classification, and we also conduct fine-tuning on the full training set for comprehensive analysis. The evaluation is on the test set of each dataset.

Baselines. For GPT-3's prompting, we utilize three different command templates for textual prompts. For each command, GPT-3 conducts 5 prompts. Then, we screened out prompts with a word count greater than 10 and pick 10 prompts for each category randomly. For SD2, we adopt

Table 1: Comparison of FLIER with the backbone of ViT-B/16 and other methods on ImageNet for fine-tuning on the full training set.

Models	FD-CLIP	CLIP-finetune	**FLIER**
Accuracy	84.94	85.67	**87.08**

DPM as the sampler and UNet as the backbone of the forward process. Additionally, we set the batch size as 2, time step in the forward process as 50 and downsampling factor as 8. With the above settings, we sample and generate low-dimensional latent representations and images via SD2. For the implementation of CLIP, when comparing to fine-tuning CLIP directly on the whole ImageNet, we utilize ViT-B/16(Dosovitskiy et al. 2020) as the backbone. When conducting few-shot experiments, we adopt ResNet50(He et al. 2016) (RN50) as the backbone of the image encoder and its aligned transformer(Vaswani et al. 2017) as the textual encoder. For fine-tuning strategies, we adopt EMA in evaluation with its momentum factor of 0.9998 and preserve model's original accuracy as well. We report the better one of two results. Also, we apply scaling, random crop, rotation and color jitter to the image for data augmentation. In the LLRD, we pick the base learning rate from 0.00005 to 0.0001 and the default increase factor of LLDR is 0.7. We adopt AdamW optimizer with the initial learning rate of 0.0001 using a cosine scheduler. The initial learning rate is modified slightly in the experiments. When training, we normally use the batch size of 64 for 40 epochs, but due to the few-shot setting and the learning rate scheduler, we need to adjust the batch size smaller in several experiments. And for few-shot experiments like 1-shot and 2-shot, we set the number of epochs larger to make model converge. We conduct all experiments with two NVIDIA GeForce RTX 3090 GPUs.

Performance of FLIER with baseline methods

Performance on ImageNet. To evaluate the vision classification ability of FLIER, We compare it with CLIP-finetune which fine-tunes CLIP(Dong et al. 2022) on ImageNet's training set. We train the models on full training set of ImageNet dataset to evaluate FLIER's performance on general sufficient data scene. As shown in Table 1, FLIER achieves the better performance than CLIP-finetune with the accuracy of 87.08%, surpassing that by 1.41%, respectively. The results show that FLIER could learn better visual representation with the help of latent representations sampled by SD2.

For few-shot classification on ImageNet, we compare FLIER with other previous CLIP-based methods, including CLIP(Radford et al. 2021), CoOp(Zhou et al. 2022b), CLIP-Adapter(Gao et al. 2023), Tip-Adapter-F(Zhang et al. 2022) and CaFo(Zhang et al. 2023). We adopt pre-trained CLIP image encoders based on RN50 backbone. The results in Figure 2(a) show that FLIER performs surprisingly well on ImageNet in few-shot experiments, especially 4-shot, 8-shot and 16-shot. It also shows some advantages compared to the other methods in 1-shot and 2-shot setting. In Table 2, FLIER achieves the highest accuracy on the test set with
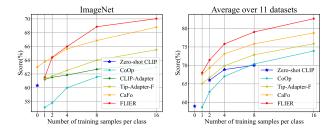

Figure 2: Average performance results on 11 datasets and performance on ImageNet of few-shot learning. FLIER with RN50 outperforms SOTA for 2, 4, 8 and 16 shot settings on ImageNet and for average accuracy on 11 datasets.

Table 2: Comparison of FLIER and other methods with RN50 on ImageNet under few-shot setting.

Shot	1	2	4	8	16
Lp-CLIP	22.17	31.90	41.20	49.52	56.13
CoOp	57.15	57.81	59.99	61.56	62.95
CLIP-Adapter	61.20	61.52	61.84	62.68	63.59
Tip-Adapter-F	61.32	61.69	62.52	64.00	65.51
CaFo	**63.80**	64.34	65.64	66.86	68.79
FLIER	61.51	**64.43**	**65.98**	**68.86**	**70.03**

(70.03%, 68.86%) under 16-shot and 8-shot setting, surpassing the other baseline models by (1.24%, 2.00%), (4.52%, 4.86%), (6.44%, 6.18%), (7.08%, 7.30%) and (13.90%, 19.34%) respectively. It is remarkable that FLIER in 8-shot setting achieves even higher accuracy (68.86%) than previous SOTA in 16-shot setting (68.79%). Under 1-shot and 2-shot setting, FLIER is still superior to other methods: Linear-probe CLIP (Lp-CLIP), CoOp, CLIP-Adapter and Tip-Adapter-F. But CaFo yields a competitive performance in 1-shot setting. Due to the large data required for fine-tuning, FLIER is relatively strong when the number of shots is larger or equal to 2. In addition, we analyze the efficiency of CaFo and FLIER in ViT-B/16 for their training time and accuracy. The results in Table 3 shows that FLIER achieves a higher accuracy 76.70% with 156 minutes of training on a single NVIDIA GeForce RTX 3090 GPU for 40 epochs, each epoch takes almost the same time as CaFo. Also, we compare the flops and parameters of FLIER with CLIP,. The results in Table 4 shows that FLIER only add 0.77M parameters in the architecture to CLIP's image encoder, indicating that FLIER achieves excellent performance with a small increase of parameters.

Performance on Other Benchmark Datasets. To analyze the ability of FLIER comprehensively, we conduct experiments of FLIER on other 10 datasets and calculate the average results of 11 datasets with ImageNet. In Figure 2(b), for the average accuracy on the 11 datasets, the accuracy of FLIER is higher than previous SOTA, CaFo with an increase of 0.46%, 2.14%, 2.67%, 3.21% and 3.87% under 1-shot, 2-shot, 4-shot, 8-shot and 16-shot settings, respectively. The few-shot results on other 10 datasets are presented in Fig-

Table 3: Comparison of time efficiency and accuracy for FLIER and CaFo on ImageNet under 16-shot setting with a single NVIDIA GeForce RTX 3090 GPU with ViT-B/16.

Models	Epoch	Time	Accuracy	Gain
Zero-shot CLIP	0	0	60.33	-
CaFo	**20**	**1h38min**	74.48	+14.15
FLIER	40	2h36min	**76.70**	**+14.72**

Table 4: Flops and parameters of FLIER and CLIP.

Models	Flops	Parameter
CLIP	17083.66 M	82.46 M
FLIER (image encoder)	11271.12 M	58.03 M
FLIER (latent encoder)	0.96 M	0.77 M

ure 3. Overall, our FLIER outperforms all CLIP-based methods under all few-shot settings on 10 datasets. Of particular note is the performance of FLIER on FGVCAircraft, FLIER increases an accuracy of 9.51%, 11.28% and 14.64% than SOTA's accuracy in 4-shot, 8-shot and 16-shot, respectively. Although FLIER shows a slight average improvement over the previous SOTA in the 1-shot setting, it outperforms the SOTA by 1.33% on the Flowers102 dataset and 2.37% on the DTD dataset. Another surprising result is that FLIER demonstrates superior performance in the 8-shot and 16-shot settings, averaging 3.21% and 3.87%, outperforming other methods by a large margin. FLIER also performs well in the 4-shot setting, with its accuracy surpassing that of all other baseline methods on Caltech101 and Food101 datasets, even outperforming their accuracy in the 16-shot setting.

Data Domain Generalization. The ability to generalize to out-of-distribution data is important for deep learning models in real-world scenes and practical applications. We evaluate the domain generalization ability of FLIER by training on ImageNet and testing on ImageNet-V2(Recht et al. 2019) and ImageNet-Sketch(Hendrycks et al. 2021), which contain the same categories with ImageNet. In Table 5, with latent encoder and prior knowledge from SD2, FLIER surpasses previous SOTA on ImageNet-V2 and ImageNet-Sketch by (1.05%, 2.01%) with RN50 backbone and (1.63%, 0.99%) with ViT-B/16 backbone. The results also demonstrate that FLIER with ViT backbone outperforms ResNet with a large margin.

Ablation studies

Generated Images via Stable Diffusion 2. We conduct ablations to observe whether the generated images from SD2 help improve the performance of CLIP-finetune. We conduct the experiments both on the whole dataset and the few-shot setting. In Table 6, the performance of CLIP-finetune with generated images achieves a higher top-1 accuracy than CLIP-finetune, but a lower top-5 accuracy. In the few-shot results, CLIP-finetune with generated images show better performance than CLIP-finetune slightly. Overall, CLIP-

Table 5: Data Domain Generalization. We train the methods on ImageNet and test on ImageNet-V2/Sketch with RN50.

Datasets	Source	Target	
	ImageNet	-V2	-Sketch
Zero-shot CLIP	60.33	53.27	35.44
CoOp	62.95	54.58	31.40
CLIP-Adapter	63.59	55.69	35.68
Tip-Adapter-F	65.51	57.11	36.00
CaFo(RN50)	68.79	57.99	39.43
CaFo(ViT-B/16)	74.48	66.33	49.10
FLIER(RN50)	70.03	59.04	41.44
FLIER(ViT-B/16)	**76.70**	**67.96**	**50.09**

Table 6: Ablations on generated images on ImageNet with RN50. CLIP-AugData represents CLIP with generated data.

Models	top-1-acc	top-5-acc	shot-1	shot-2
CLIP-finetune	85.67	97.26	47.05	53.31
CLIP-AugData	85.82	97.24	47.30	54.72
FLIER	**87.08**	**98.65**	**61.51**	**64.43**
Models	shot-4	shot-8	shot-16	-
CLIP-finetune	61.71	66.85	68.35	-
CLIP-AugData	62.22	66.89	68.42	-
FLIER	**65.98**	**68.86**	**70.03**	-

finetune with generated images does not show obvious better performance than CLIP-finetune, which proves the effectiveness of another module of low-dimensional latent representations learned by SD2.

Low-dimensional Latent Representations. After excluding the effectiveness of generated images, we conduct ablations on low-dimensional latent representations. Similar to the experiments in ablation of generated images, we perform ablations on whole ImageNet. In Table 6, FLIER exceeds CLIP-finetune and CLIP-AugData obviously with an increase of 1.41%, 1.26% in top-1 accuracy and 1.39%, 1.41% in top-5 accuracy. This demonstrates the effectiveness of the general vision representation learning in FLIER. In addition, FLIER is a strong few-shot learner than CLIP-finetune, outperforming that by an increase of 17.46%, 13.10%, 4.27%, 2.01% and 1.68% in accuracy under 1-shot,

Table 7: Ablations on backbone of CLIP on ImageNet.

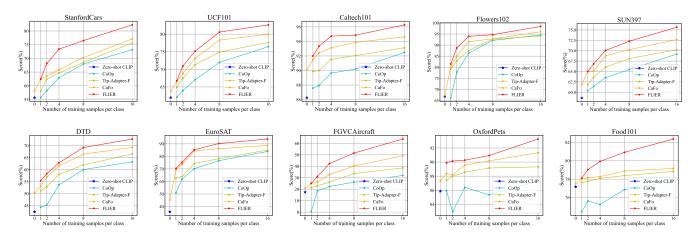| (16-shot)	RN50	RN101	ViT-B/32	ViT-B/16
Zero-shot CLIP	60.33	62.53	63.80	68.73
CoOp	62.95	66.60	66.85	71.92
CLIP-Adapter	63.59	65.39	66.19	71.13
Tip-Adapter-F	65.51	68.56	68.65	73.69
CaFo	68.79	70.86	70.82	74.48
FLIER	**70.03**	**72.17**	**71.46**	**76.70**

Figure 3: Comparison of FLIER and other methods with backbone of RN50 on 10 datasets.

Table 8: Ablations on latent factor on 16-shot ImageNet.

Latent factor	0.1	0.3	0.5	0.7	0.9
FLIER(ViT-B/16)	76.47	76.59	**76.70**	76.51	76.23

Table 9: Ablations on stages' order on 16-shot ImageNet.

Shot	1	2	4	8	16
FLIER(Joint 1st)	65.21	67.09	68.71	72.72	76.68
FLIER	65.22	67.11	68.63	72.72	76.70

2-shot, 4-shot, 8-shot and 16-shot, respectively. By two ablation studies, we conclude that additional data is not the main reason for FLIER's superior performance, instead, the low-dimensional latent representations work in FLIER.

CLIP's Visual Encoders. In FLIER, we conducted experiments using various visual encoders of CLIP to demonstrate the versatility of FLIER across different backbones. As shown in Table 7, the performances of FLIER with ViT-B/16 are higher than other visual backbones with a large margin. In addition, FLIER outperforms all other baselines across different backbones, indicating the effectiveness of FLIER with different network architectures.

Latent Factor. The latent factor in FLIER is to control the effect that the latent encoder contributes to the whole model. We explore the model's performance with different latent factors from 0.1 to 0.9 ($\alpha = 0.1, 0.3, 0.5, 0.7, 0.9$) on ImageNet under 16-shot setting in the backbone of ViT-B/16. From Table 8, we observe that neither too large nor too small latent factors contribute to better few-shot performance. Since CLIP's image encoder is trained twice with both generated images and training images, a larger latent factor might not lead to a thorough fine-tuning of the image encoder, which affects the performance. On the contrary, when the latent factor is too small, the contributions from the latent encoder are significantly reduced, causing the model to behave close to the original CLIP-finetune. When the latent factor is 0.5, the model's performance in experiments is relatively higher than that in other experiments, demonstrating that FLIER benefits the most from the relative average contribution from the latent encoder and image encoder.

Order of two stages. We swap the order of two stages to explore whether the stages' order would influence the performance of FLIER. We conduct the experiment on FLIER with the backbone of ViT-B/16 on the setting of 16-shot on ImageNet. FLIER(Joint 1st) represents that we train the joint training phrase first and train the image encoder later in one FLIER's training stage. From Table 9, there are no obvious differences between two different orders, indicating the order does not affect the performance of FLIER.

Images-only architecture with an image-to-image generation module

Taking into account the potential benefits of textual information within the FLIER framework, we replace the text-to-image generation module of SD2 with an image-to-image generation module, setting the corresponding prompt to an empty string. In this framework, the latent representation generated by the image-to-image module is aligned with the shape of CLIP's image encoder through a trainable deconvolution layer. The image-to-image sampler, deconvolution layer, and image encoder were trained concurrently. Experimental results indicate that, after modifying the framework to image-to-image generation, the model fails to converge and yields very low accuracy on the test set. We hypothesize that the upsampling performed by the deconvolution layer within the pipeline, which inputs the latent representation into the image encoder, significantly degrades the quality of the latent representation. Also, the ViT with a fixed patch size and pre-trained weights produces a large number of redundant and non-informative tokens during image tokenization, leading to poor training performance. These findings support the use of diffusion models of FLIER over others.

Limitations

While we believe FLIER to be a robust few-shot learner, we have not conducted experiments on professional application datasets. Therefore, the performance of FLIER in real-world scene applications remains uncertain. In the future, we aim to conduct additional experiments to evaluate FLIER's practical applicability. In addition, due to the characteristics of fine-tuning and few-shot setting, FLIER's performance in 1-shot setting is not as strong as more shots setting.

Conclusion

In this paper, we explored the feasibility of low-dimensional latent representations generated from image synthesis tasks for image recognition and few-shot learning. Leveraging GPT-3 and SD2, we obtained latent representations, which were then embedded into a vision-language model, CLIP, through the latent encoder architecture of FLIER. To further investigate the auxiliary effects of latent representations on FLIER, we introduced latent factors to quantitatively explore the performance of FLIER with different contributions of latent representations. In the few-shot experiments across 11 datasets, FLIER has demonstrated SOTA performance for most tasks. In the future, we plan to investigate the possibility of constructing an independent backbone, possibly based on ViT, with latent representations as the core module. This backbone could potentially be versatile and applicable across various tasks.

We declare that we will release source code upon acceptance of the paper.

References

Bao, H.; Dong, L.; Piao, S.; and Wei, F. 2021. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*.

Benaim, S.; and Wolf, L. 2018. One-shot unsupervised cross domain translation. *advances in neural information processing systems*, 31.

Bossard, L.; Guillaumin, M.; and Van Gool, L. 2014. Food-101–mining discriminative components with random forests. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, 446–461. Springer.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Cai, Q.; Pan, Y.; Yao, T.; Yan, C.; and Mei, T. 2018. Memory matching networks for one-shot image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4080–4088.

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9650–9660.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Chen, X.; Ding, M.; Wang, X.; Xin, Y.; Mo, S.; Wang, Y.; Han, S.; Luo, P.; Zeng, G.; and Wang, J. 2024. Context autoencoder for self-supervised representation learning. *International Journal of Computer Vision*, 132(1): 208–223.

Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; and Vedaldi, A. 2014. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3606–3613.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34: 8780–8794.

Dong, X.; Bao, J.; Zhang, T.; Chen, D.; Gu, S.; Zhang, W.; Yuan, L.; Chen, D.; Wen, F.; and Yu, N. 2022. Clip itself is a strong fine-tuner: Achieving 85.7% and 88.0% top-1 accuracy with vit-b and vit-l on imagenet. *arXiv preprint arXiv:2212.06138*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Douze, M.; Szlam, A.; Hariharan, B.; and Jégou, H. 2018. Low-shot learning with large-scale diffusion. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3349–3358.

Fei-Fei, L.; Fergus, R.; and Perona, P. 2004. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, 178–178. IEEE.

Fürst, A.; Rumetshofer, E.; Lehner, J.; Tran, V. T.; Tang, F.; Ramsauer, H.; Kreil, D.; Kopp, M.; Klambauer, G.; Bitto, A.; et al. 2022. Cloob: Modern hopfield networks with infoloob outperform clip. *Advances in neural information processing systems*, 35: 20450–20468.

Gao, H.; Shou, Z.; Zareian, A.; Zhang, H.; and Chang, S.-F. 2018. Low-shot learning via covariance-preserving adversarial augmentation networks. *Advances in Neural Information Processing Systems*, 31.

Gao, P.; Geng, S.; Zhang, R.; Ma, T.; Fang, R.; Zhang, Y.; Li, H.; and Qiao, Y. 2023. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 1–15.

Gao, P.; Jiang, Z.; You, H.; Lu, P.; Hoi, S. C.; Wang, X.; and Li, H. 2019. Dynamic fusion with intra-and inter-modality

attention flow for visual question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6639–6648.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16000–16009.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9729–9738.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Helber, P.; Bischke, B.; Dengel, A.; and Borth, D. 2019. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7): 2217–2226.

Henaff, O. 2020. Data-efficient image recognition with contrastive predictive coding. In *International conference on machine learning*, 4182–4192. PMLR.

Hendrycks, D.; Zhao, K.; Basart, S.; Steinhardt, J.; and Song, D. 2021. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15262–15271.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.-T.; Parekh, Z.; Pham, H.; Le, Q.; Sung, Y.-H.; Li, Z.; and Duerig, T. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, 4904–4916. PMLR.

Kim, J.-H.; Jun, J.; and Zhang, B.-T. 2018. Bilinear attention networks. *Advances in neural information processing systems*, 31.

Kingma, D.; Salimans, T.; Poole, B.; and Ho, J. 2021. Variational diffusion models. *Advances in neural information processing systems*, 34: 21696–21707.

Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. *stat*, 1050: 1.

Kozerawski, J.; and Turk, M. 2018. Clear: Cumulative learning for one-shot one-class image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3446–3455.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, 554–561.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

Li, Y.; Liang, F.; Zhao, L.; Cui, Y.; Ouyang, W.; Shao, J.; Yu, F.; and Yan, J. 2021. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv preprint arXiv:2110.05208*.

Liu, Y.; Lee, J.; Park, M.; Kim, S.; Yang, E.; Hwang, S. J.; and Yang, Y. 2018. Learning to propagate labels: Transductive propagation network for few-shot learning. *arXiv preprint arXiv:1805.10002*.

Lu, C.; Zhou, Y.; Bao, F.; Chen, J.; Li, C.; and Zhu, J. 2022. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35: 5775–5787.

Lu, J.; Batra, D.; Parikh, D.; and Lee, S. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32.

Maji, S.; Rahtu, E.; Kannala, J.; Blaschko, M.; and Vedaldi, A. 2013. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*.

Nilsback, M.-E.; and Zisserman, A. 2008. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, 722–729. IEEE.

Parkhi, O. M.; Vedaldi, A.; Zisserman, A.; and Jawahar, C. 2012. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, 3498–3505. IEEE.

Pfister, T.; Charles, J.; and Zisserman, A. 2014. Domainadaptive discriminative one-shot learning of gestures. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, 814–829. Springer.

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

Ramalho, T.; and Garnelo, M. 2019. Adaptive posterior learning: few-shot learning with a surprise-based memory module. *arXiv preprint arXiv:1902.02527*.

Ramesh, A.; Pavlov, M.; Goh, G.; Gray, S.; Voss, C.; Radford, A.; Chen, M.; and Sutskever, I. 2021. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, 8821–8831. PMLR.

Ravi, S.; and Larochelle, H. 2016. Optimization as a model for few-shot learning. In *International conference on learning representations*.

Recht, B.; Roelofs, R.; Schmidt, L.; and Shankar, V. 2019. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, 5389–5400. PMLR.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent

diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, 234–241. Springer.

Rusu, A. A.; Rao, D.; Sygnowski, J.; Vinyals, O.; Pascanu, R.; Osindero, S.; and Hadsell, R. 2018. Meta-learning with latent embedding optimization. *arXiv preprint arXiv:1807.05960*.

Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2256–2265. PMLR.

Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2020. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.

Soomro, K.; Zamir, A. R.; and Shah, M. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Tan, H.; and Bansal, M. 2019. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*.

Tsai, Y.-H. H.; and Salakhutdinov, R. 2017. Improving one-shot learning through fusing side information. *arXiv preprint arXiv:1710.08347*.

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

Wang, Y.; Yao, Q.; Kwok, J. T.; and Ni, L. M. 2020. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, 53(3): 1–34.

Xiao, J.; Hays, J.; Ehinger, K. A.; Oliva, A.; and Torralba, A. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, 3485–3492. IEEE.

Yu, M.; Guo, X.; Yi, J.; Chang, S.; Potdar, S.; Cheng, Y.; Tesauro, G.; Wang, H.; and Zhou, B. 2018. Diverse few-shot text classification with multiple metrics. *arXiv preprint arXiv:1805.07513*.

Yu, Z.; Yu, J.; Cui, Y.; Tao, D.; and Tian, Q. 2019. Deep modular co-attention networks for visual question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6281–6290.

Zhang, R.; Hu, X.; Li, B.; Huang, S.; Deng, H.; Qiao, Y.; Gao, P.; and Li, H. 2023. Prompt, generate, then cache: Cascade of foundation models makes strong few-shot learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15211–15222.

Zhang, R.; Zhang, W.; Fang, R.; Gao, P.; Li, K.; Dai, J.; Qiao, Y.; and Li, H. 2022. Tip-adapter: Training-free adaption of clip for few-shot classification. In *European Conference on Computer Vision*, 493–510. Springer.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022a. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16816–16825.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022b. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9): 2337–2348.