This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Bayesian Exploration of Pre-trained Models for Low-shot Image Classification

Yibo Miao[1], Yu Lei[1], Feng Zhou[2*], Zhijie Deng[1*]
[1]Qing Yuan Research Institute, SEIEE, Shanghai Jiao Tong University
[2]Center for Applied Statistics and School of Statistics, Renmin University of China
{miaoyibo, tony-lei, zhijied}@sjtu.edu.cn, feng.zhou@ruc.edu.cn

Abstract

Low-shot image classification is a fundamental task in computer vision, and the emergence of large-scale vision-language models such as CLIP has greatly advanced the forefront of research in this field. However, most existing CLIP-based methods lack the flexibility to effectively incorporate other pre-trained models that encompass knowledge distinct from CLIP. To bridge the gap, this work proposes a simple and effective probabilistic model ensemble framework based on Gaussian processes, which have previously demonstrated remarkable efficacy in processing small data. We achieve the integration of prior knowledge by specifying the mean function with CLIP and the kernel function with an ensemble of deep kernels built upon various pre-trained models. By regressing the classification label directly, our framework enables analytical inference, straightforward uncertainty quantification, and principled hyper-parameter tuning. Through extensive experiments on standard benchmarks, we demonstrate that our method consistently outperforms competitive ensemble baselines regarding predictive performance. Additionally, we assess the robustness of our method and the quality of the yielded uncertainty estimates on out-of-distribution datasets. We also illustrate that our method, despite relying on label regression, still enjoys superior model calibration compared to most deterministic baselines.

1. Introduction

The past few years have witnessed the trend of training large-scale foundation models to serve as infrastructures for processing images, texts, and multi-modal data [3, 7, 12, 20, 24, 44]. The increasing availability of off-the-shelf pre-trained models is changing the standard practice for solving specific downstream tasks for AI practitioners. One fundamental application in vision is adapting pre-trained models for low-shot image classification. This eliminates the need for massive labeled data as in traditional cases, helps initiate the data annotation process, and supports the construction of complex recognition systems, among other advantages.

Fine-tuning and linear probing are typical approaches for pre-trained models-based low-shot image classification [5, 7, 8, 27]. Recently, vision-language models, e.g., CLIP [44], have significantly advanced zero-shot classification where the image and semantics of interest are projected into a structured hidden space for nearest neighbor-based classification. Nevertheless, the few-shot CLIP with linear probing shows inferior results [44]. To address this, researchers have put considerable effort into developing novel CLIP-based few-shot learning pipelines involving techniques such as prompting learning [59], image-guided prompt generation [43, 58], adapter tuning [15, 56], etc.

Despite relatively good results, existing CLIP-based methods usually lose the flexibility to incorporate other pre-trained models that may contain complementary prior information. CaFo [57] is a seminal work that explores constructing few-shot predictors using pre-trained models other than CLIP and demonstrates outperforming effectiveness. However, the ensemble weights in CaFo are determined heuristically, and the learning requires extensive hyper-parameter tuning. Furthermore, as a deterministic method, CaFo is likely to overfit the few-shot training data and cannot provide accurate uncertainty estimates. These challenges are particularly troublesome in situations with limited data and high-risk domains.

This paper aims to assemble CLIP and other pre-trained models in a more principled probabilistic manner. Given that previous studies usually deploy a linear classification head on top of the pre-trained models, we focus on its Bayesian counterpart, i.e., a *Gaussian process* (GP) [53]. GP is an ideal model for low-shot image classification due to its effectiveness with *small* data. To incorporate prior knowledge from various pre-trained models, we suggest defining the prior kernel as a combination of deep kernels associated with various pre-trained models. Noting that the prior mean implicitly corresponds to a model that makes predictions without seeing any data, we specify it with the

*Corresponding authors.

well-performing zero-shot CLIP classifier.

Such a modeling can address overfitting and result in calibrated *post-data* uncertainty arising from posterior inference. Further, the Bayesian framework allows for the use of principled objectives for hyper-parameter tuning. For example, we can use the marginal likelihood or predictive likelihood of the GP model for hyper-parameter tuning following common practice.

We begin by assessing the predictive performance of our proposed method on standard low-shot image classification benchmarks and observe superior or competitive results compared to a variety of ensemble baselines. To evaluate the generalization capability of our method, we test the trained models on natural out-of-distribution (OOD) data and find that our method achieves outperforming results. In addition, our method has the potential to yield calibrated uncertainty estimates for OOD data. We further assess the model calibration by inspecting Expected Calibration Error (ECE) [16] and its more robust variant, Thresholded Adaptive Calibration Error (TACE) [39]. We also offer thorough ablation studies to better understand the proposed method.

2. Related Works

Zero/few-shot classification. Few-shot classification means making classifications based on a limited number of observations, and the zero-shot one requires the trained model to adapt to the new task without any observation. Meta-learning has demonstrated its potential as a viable approach for zero/few-shot learning [47, 50]. Recently, benefiting from the learning on web-scale data, large pre-trained vision-language models like CLIP have demonstrated impressive performance in zero/few-shot image classification. Since then, continual effort has been made to better adapt CLIP to downstream few-shot tasks [15, 17, 51, 56–59]. In particular, CoOp [59] optimizes a collection of learnable prompt tokens for few-shot adaptation. Tip-Adapter [56] augments the zero-shot CLIP classifier with a linear key-cache cache model to further enhance the classification performance. CaFo [57] supplements Tip-Adapter with one further linear key-cache cache model for knowledge integration. Although effective, the deterministic nature of these methods makes them tend to overfit the few-shot training data and struggle to estimate predictive uncertainty.

Pre-trained models in vision and beyond. We have witnessed the change of model architectures in vision from VGG [48] and ResNet [18] to ViT [13] and Swin Transformer [33]. The dominant learning paradigm has undergone a transformation, where pre-training models on extensive datasets and then utilizing them for downstream tasks via fine-tuning [19] has become a widespread practice. MoCo [9] and DINO [5] are recent representative pre-trained models, enjoying the ability to generate high-quality representations. Visual pre-trained models are instrumental in achieving state-of-the-art performance on diverse downstream tasks such as object detection [32], semantic segmentation [6], and so on. Recently, visual-language pre-training has achieved impressive success by learning from massive image-text pairs gathered from the internet [24, 31, 44], demonstrating astonishing performance on various downstream vision and language tasks. We have reached the consensus that pre-trained models can serve as containers of valuable prior knowledge, but a proper mechanism for effective knowledge integration is under-explored, which is alleviated by this work.

Deep Gaussian processes. GPs are a well-studied and powerful probabilistic tool in machine learning [53]. They share a deep connection with neural networks (NNs) with infinite width [23, 30, 36]. There exists an interesting correspondence among linear regression, Bayesian linear regression, and GP regression, with the last one often preferred in low-data regimes. GPs have been successfully used to solve classification problems based on approximations [37]. However, GPs built on classic kernels lack the inductive bias carried by NNs. To address this, deep kernel learning (DKL) [4, 54] has been proposed to leverage deep NNs for nonlinear data projection, which is then fed to classic kernels. In the context of few-shot learning, deterministic methods with a linear classification head face challenges of overfitting to the training set and are unable to accurately quantify uncertainty. This limitation restricts their applicability in high-risk domains. In contrast, GPs offer a viable remedy to these pathologies.

3. Preliminary

This section reviews the basics of GP regression and deep kernel learning. We use $\mathcal{D} = \{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^N$ to denote a dataset with $\boldsymbol{x}_i \in \mathcal{X} \subset \mathbb{R}^L$ and $\boldsymbol{y}_i \in \mathbb{R}^C$ as the L-dim inputs and C-dim targets respectively. Let $\mathbf{X} = \{\boldsymbol{x}_i\}_{i=1}^N$ and $\mathbf{Y} = \{\boldsymbol{y}_i\}_{i=1}^N$ represent the training data. Let $\mathbf{X}^{\text{val}}$ and $\mathbf{Y}^{\text{val}}$ represent the validation data (can be split from the training data) and $\mathbf{X}^* = \{\boldsymbol{x}_i^*\}_{i=1}^M$ represent the test data.

3.1. From Deterministic to Bayesian

To deal with the learning problem on the above dataset, it is common practice to train a deterministic model $f : \mathcal{X} \to \mathbb{R}^C$ using maximum likelihood estimation or maximum a posteriori principle. Despite effectiveness, the approach can suffer from detrimental overfitting and struggle to reason about model uncertainty appropriately. These issues are exacerbated when only limited data is available.

Practitioners can turn to Bayesian learning approaches to address such issues. In Bayesian learning, a prior distribution over model parameters is introduced, and the Bayesian posterior is (approximately) computed. Then, we compute the posterior predictive distribution to predict for a new datum, where all likely model specifications are considered.

The uncertainty can be quantified by certain statistics that capture the degree of variation in that distribution.

3.2. Gaussian Process Regression

GP regression is an extensively studied function-space Bayesian model [52]. It enjoys exact Bayesian inference and non-parametric flexibility, allowing for a high degree of freedom in kernel specification to adapt the model to various types of nonlinear data. Consequently, it is often the preferred choice for *small*- to *medium*-sized datasets.

Specifically, GP regression usually deploys a prior in the following formula:

$$f(\boldsymbol{x}) \sim \mathcal{GP}(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')), \tag{1}$$

where $m(\boldsymbol{x})$ indicates the mean function and $k(\boldsymbol{x}, \boldsymbol{x}')$ denotes the kernel (covariance) function that describes the similarity among data points. Assume additive isotropic Gaussian noise on the function output, which corresponds to a Gaussian likelihood $y(\boldsymbol{x})|f(\boldsymbol{x}) \sim \mathcal{N}(y(\boldsymbol{x}); f(\boldsymbol{x}), \sigma^2\mathbf{I})$ where σ^2 is the noise variance. The predictive distribution of the function evaluations $\boldsymbol{f}^*$ on new data points $\mathbf{X}^*$ is:

$$\boldsymbol{f}^*|\mathbf{X}^*, \mathbf{X}, \mathbf{Y} \sim \mathcal{N}(\mathbb{E}[\boldsymbol{f}^*], \mathrm{cov}(\boldsymbol{f}^*)), \tag{2}$$

where

$$\begin{aligned} \mathbb{E}[\boldsymbol{f}^*] &:= \boldsymbol{m}_{\mathbf{X}^*} + \boldsymbol{k}_{\mathbf{X}^*,\mathbf{X}}[\boldsymbol{k}_{\mathbf{X},\mathbf{X}} + \sigma^2\mathbf{I}]^{-1}(\mathbf{Y} - \boldsymbol{m}_{\mathbf{X}}), \\ \mathrm{cov}(\boldsymbol{f}^*) &:= \boldsymbol{k}_{\mathbf{X}^*,\mathbf{X}^*} - \boldsymbol{k}_{\mathbf{X}^*,\mathbf{X}}[\boldsymbol{k}_{\mathbf{X},\mathbf{X}} + \sigma^2\mathbf{I}]^{-1}\boldsymbol{k}_{\mathbf{X},\mathbf{X}^*}, \end{aligned} \tag{3}$$

$\boldsymbol{m}_{\mathbf{X}} \in \mathbb{R}^{N\times C}$ and $\boldsymbol{k}_{\mathbf{X},\mathbf{X}} \in \mathbb{R}^{N\times N}$ represent the evaluation of $m(\cdot)$ and $k(\cdot,\cdot)$ on the training data $\mathbf{X}$ respectively. Other matrices are defined similarly. Unlike parametric models such as NNs, GP makes predictions for new data by referring to the training samples, similar to how humans approach the task.

This model can be readily adapted to tackle classification problems by treating the one-hot labels as regression targets, which is known as the label regression [29, 30, 42]. The label regression design enables analytical expressions for both evidence and posterior, making the classifier computationally efficient and easy to implement.

The GP regression also offers analytical objectives for tuning parameters (denoted as $\boldsymbol{\alpha}$; including σ^2 and others in the definition of m and k). One typical choice is the log marginal likelihood:

$$\begin{aligned} \log p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\alpha}) \propto -[\mathrm{trace}((\mathbf{Y} - \boldsymbol{m}_{\mathbf{X}})^\top(\boldsymbol{k}_{\mathbf{X},\mathbf{X}} \\ + \sigma^2\mathbf{I})^{-1}(\mathbf{Y} - \boldsymbol{m}_{\mathbf{X}})) + C\log|\boldsymbol{k}_{\mathbf{X},\mathbf{X}} + \sigma^2\mathbf{I}|], \end{aligned} \tag{4}$$

which corresponds to the summation of the log marginal likelihood of C independent 1-dim GP regressions. Yet, it is shown that this objective can be negatively correlated with the generalization [25, 34]. Given this, a more proper objective can be $\log p(\mathbf{Y}^{\mathrm{val}}|\mathbf{X}^{\mathrm{val}}, \mathbf{X}, \mathbf{Y}, \boldsymbol{\alpha})$, i.e., the predictive likelihood on extra validation data $(\mathbf{X}^{\mathrm{val}}, \mathbf{Y}^{\mathrm{val}})$. It also takes the form of Gaussian log densities.

3.3. Deep Kernel Learning

In DKL, a $\boldsymbol{\theta}$-parameterized deep NN $g_{\boldsymbol{\theta}} : \mathcal{X} \to \mathbb{R}^D$ is typically used to transform the input data $\boldsymbol{x}$ into hidden features $g_{\boldsymbol{\theta}}(\boldsymbol{x})$. The kernel is then defined as:

$$k(\boldsymbol{x}, \boldsymbol{x}') := \tilde{k}(g_{\boldsymbol{\theta}}(\boldsymbol{x}), g_{\boldsymbol{\theta}}(\boldsymbol{x}')), \tag{5}$$

where $\tilde{k}$ is a base kernel, such as the popular radial basis function (RBF) kernel or polynomial kernel.

To make the NN parameters better suited for the data at hand, DKL treats them as hyper-parameters of the GP model and optimizes them to maximize the marginal likelihood. However, the large number of hyper-parameters makes the optimization time-consuming and increases the risk of overfitting [40]. It can even underperform a standard deterministic NN in some toy cases.

4. Methodology

This section explores a Bayesian approach to assemble CLIP with other pre-trained models for low-shot image classification. Given the discussion above, we take the GP regression as the modeling framework. We then elaborate on how to integrate various pre-trained models into it. We provide an overview of our method in Fig. 1.

4.1. Design of Kernel

Utilizing an NN-based feature extractor to define the kernel function aids to incorporate informative inductive bias into GP, which is essential for processing complex data such as images and texts. However, conventional approaches like DKL suffer from the pathology that all involved NN parameters are required to be carefully tuned. Considering that pre-trained models can yield representations that are generally applicable to a wide range of applications, we propose to alternatively use pre-trained models to define deep kernels and then perform an adaptive combination. By doing this, the number of hyper-parameters in the GP is reduced significantly, and the prior knowledge encoded by various pre-trained models is effectively integrated.

Specifically, assuming access to K pre-trained models $g^{(i)} : \mathcal{X} \to \mathbb{R}^{D^{(i)}}, i = 1, \ldots, K$,[1] we define the following independent kernels:

$$k^{(i)}(\boldsymbol{x}, \boldsymbol{x}') := \tilde{k}(\boldsymbol{l}^{(i)} \circ g^{(i)}(\boldsymbol{x}), \boldsymbol{l}^{(i)} \circ g^{(i)}(\boldsymbol{x}')), \tag{6}$$

where $\circ$ denotes the element-wise product and $\boldsymbol{l}^{(i)} \in \mathbb{R}_+^{D^{(i)}}$ is a learnable vector used to boost flexibility, e.g., when the base kernel $\tilde{k}$ is the RBF kernel, $\boldsymbol{l}^{(i)}$ defines the learnable length-scales for it. We can also enforce a constraint where

[1] We omit the dependency of these models on their parameters because we regard them as fixed models and do not perform fine-tuning. We assume an L^2 normalization at the end of each model unless specified otherwise.

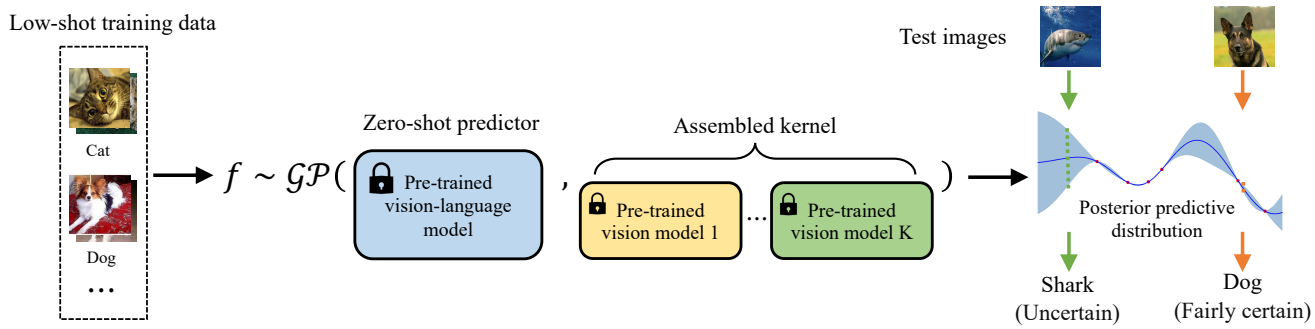


Figure 1. Overview of our method. We leverage a GP regressor to tackle the low-shot image classification problem. To integrate knowledge from CLIP and other pre-trained models, we use them to specify the GP mean and kernel. The label is determined by the mean, and the uncertainty estimate is determined by the variance.

all elements in $\boldsymbol{l}^{(i)}$ have the same value, and the final learning outcomes are slightly impacted. By summing up these kernels, we get the final kernel:

$$k(\boldsymbol{x}, \boldsymbol{x}') := \sum_{i=1}^{K} k^{(i)}(\boldsymbol{x}, \boldsymbol{x}'). \tag{7}$$

The learnable hyper-parameters enable an easy, automatic adaptation of the kernel to specific data.

4.2. Design of Mean

In essence, the prior mean $m(\cdot)$ refers to a function making predictions before seeing any data, i.e., a zero-shot predictor. Traditionally, $m(\cdot)$ is set to zero for simplicity. However, as shown in Sec. 5.5, this can lead to poor generalization performance in low-shot image classification tasks in practice. This suggests that it is necessary to incorporate effective prior knowledge of $m(\cdot)$ into the GP.

Interestingly, a similar phenomenon has been reported in the literature, where the linear probe CLIP using few-shot data performs much worse than zero-shot CLIP [44]. This is because the knowledge in the zero-shot CLIP classifier has not been effectively integrated into the few-shot learners.

With these insights, we make a simple yet significant improvement to our GP model. We set the mean function $m(\cdot)$ to the zero-shot linear classifier in CLIP, which has demonstrated strong performance. Concretely, let $g : \mathcal{X} \to \mathbb{R}^D$ denotes CLIP's image encoder, and $\boldsymbol{w} \in \mathbb{R}^{D \times C}$ denotes the weight of the zero-shot linear classifier composed of embeddings of the text descriptions of the C classes of interest. Our prior mean takes the following form:

$$m(\boldsymbol{x}) := \gamma \, \mathrm{softmax}(\tau g(\boldsymbol{x})^\top \boldsymbol{w}), \tag{8}$$

where $\tau, \gamma \in \mathbb{R}_+$ denote the introduced learnable temperature and scale respectively. Notably, we use a softmax operation to obtain classification probabilities directly because we formulate the classification problem as a regression one.

Algorithm 1 Leverage Gaussian processes to assemble pre-trained models for low-shot image classification

Input: Number of optimization steps T, training data $\mathbf{X}$, $\mathbf{Y}$, validation data $\mathbf{X}^{\text{val}}$, $\mathbf{Y}^{\text{val}}$, test data $\mathbf{X}^*$, hyper-parameters $\boldsymbol{\alpha}$.
Output: Predictions ($\mathbf{Y}^*$) of $\mathbf{X}^*$ and $\mathrm{cov}(\boldsymbol{f}^*)$.
for $t = 1 \to T$ **do**
  Obtain $\mathbb{E}[\boldsymbol{f}^{\text{val}}]$ and $\mathrm{cov}(\boldsymbol{f}^{\text{val}})$ of $\mathbf{X}^{\text{val}}$ via Eq. (3);
  Calculate $\log p(\mathbf{Y}^{\text{val}} | \mathbf{X}^{\text{val}}, \mathbf{X}, \mathbf{Y}, \boldsymbol{\alpha})$ and estimate its gradients w.r.t. $\boldsymbol{\alpha}$;
  Update $\boldsymbol{\alpha}$ by one-step gradient ascent;
Obtain $\mathbb{E}[\boldsymbol{f}^*]$ and $\mathrm{cov}(\boldsymbol{f}^*)$ of $\mathbf{X}^*$ via Eq. (3);
$\mathbf{Y}^* = \mathrm{argmax}(\mathbb{E}[\boldsymbol{f}^*])$;

4.3. Learning

Using the classification likelihood for data fitting is also viable. However, doing so naturally disrupts conjugacy, leading to the inability to estimate the posterior in a closed form [1, 53]. Therefore, we advocate label regression for its computational efficiency and ease of implementation. It also allows us to revert to analytical expressions for both the evidence and the posterior.

One extra merit of label regression is that it enables the tractable marginalization of data likelihood, so we can perform hyper-parameter tuning more easily. Let $\boldsymbol{\alpha} := \{\sigma^2, \boldsymbol{l}^{(1)}, \ldots, \boldsymbol{l}^{(K)}, \tau, \gamma\}$ denote all hyper-parameters in our method. We optimize them by maximizing the aforementioned log marginal likelihood or log predictive likelihood to make them suitable for the data. In the low-shot learning scenario, the dataset size is small, allowing us to compute the kernel matrix, its inversion, and its determinant with minimal cost. Using an Adam optimizer [26], convergence is usually rapid, typically within 100 optimization steps. We depict the overall algorithmic procedure in Algorithm 1.

Shot	1	2	4	8	16
Ens-LP	40.25 ± 0.09	49.79 ± 0.07	57.42 ± 0.06	62.28 ± 0.09	66.31 ± 0.13
Ens-LP†	61.77 ± 0.13	64.10 ± 0.35	65.89 ± 0.33	67.59 ± 0.06	69.83 ± 0.27
Ens-CaFo	62.09 ± 0.13	63.67 ± 0.19	64.96 ± 0.06	66.57 ± 0.42	68.78 ± 0.25
Ours	$\mathbf{63.07 \pm 0.07}$	$\mathbf{65.17 \pm 0.23}$	$\mathbf{67.50 \pm 0.06}$	$\mathbf{69.31 \pm 0.08}$	$\mathbf{70.77 \pm 0.07}$

Table 1. Comparison with ensemble baselines of low-shot classification accuracy (%) on ImageNet.

4.4. Uncertainty Quantification

As per convention, we utilize the diagonal elements of $\mathrm{cov}(\boldsymbol{f}^*)$ (outlined in Eq. (3)) to quantify the predictive uncertainty of our model on the test data points. This information enables us to refrain from making predictions on data with high uncertainty and, instead, implement other conservative fallback strategies to handle such situations. Moreover, we can leverage this information to identify OOD samples since they typically exhibit greater uncertainty than in-distribution data.

5. Experiments

We first demonstrate that our method achieves competitive low-shot performance on diverse and prevalent benchmarks. Subsequently, we illustrate how our uncertainty estimates can identify OOD samples and validate the calibration of the learning outcomes. We also conduct ablation studies on our method and provide analyses.

5.1. Experimental Setup

Datasets. Following CaFo [57], we conduct experiments on image classification datasets including ImageNet [11] and 10 other widely-used ones: StanfordCars [28], UCF101 [49], Caltech101 [14], Flowers102 [38], SUN397 [55], DTD [10], EuroSAT [21], FGVCAircraft [35], OxfordPets [41], and Food101 [2]. We follow CaFo to train the model with 1, 2, 4, 8, and 16 shots of training data and test on the entire test set.

Pre-trained models. Unless specified otherwise, we use the ResNet-50 version of CLIP. Besides, we consider the model trained by MoCo with ResNet-50 architecture, and that trained by DINO with ResNet-50 architecture for ensemble due to their popularity. We clarify that other pre-trained models are readily applicable to our framework.

Baselines. To validate that our ensemble strategy is non-trivial, we build three baselines for comparison: (1) Ens-LP, short for the ensemble of linear probing, where we apply linear probing to each pre-trained model and take the average of their output probabilities for prediction, (2) Ens-LP†, where the zero-shot CLIP classifier is further integrated into Ens-LP, and (3) Ens-CaFo, where we generalize the original CaFo approach to assemble multiple pre-trained models by fusing logits. Notably, the original CaFo approach uses images generated by DALL·E [45] for data augmentation. We do not use this strategy for all results reported in our paper.

Training protocols. For the hyper-parameters, we initialize the noise variance $\sigma^2 = 0.01$, the scale $\gamma = 1$, and the temperature $\tau = 100$. The learnable length-scales $\boldsymbol{l}$ of the kernel are randomly initialized. We can also constrain all elements in $\boldsymbol{l}$ to have the same value, and the results are slightly impacted.[2] Notably, since optimizing hyper-parameters using predictive likelihood requires a validation split, which is not feasible under 1-shot setting of ImageNet, we instead utilize marginal likelihood to optimize the hyper-parameters in that case. When using the predictive likelihood, there is an equal 1:1 ratio between the training and validation splits. On other datasets, following CaFo [57], we tune the hyper-parameters by the official validation sets. We perform hyper-parameter optimization for 100 steps with an Adam optimizer with a learning rate of 0.01 (a cosine decay is adopted). The optimization is low-cost, e.g., only requiring about 4 minutes on a single RTX-3090 under the 16-shot ImageNet setting. We follow CaFo to construct the zero-shot CLIP classifier. We report the average results over three random runs.

5.2. Predictive Performance

We first evaluate the low-shot classification performance on the ImageNet benchmark. The results are presented in Tab. 1. As shown, our method outperforms other baselines with clear margins. The less favorable results of the baselines underscore the inherent challenges in amalgamating knowledge from multiple pre-trained models. The merits of our method are more prominent for medium-sized training data (e.g., the 4 and 8 shots). It is worth noting that the performance difference between Ens-LP and Ens-LP† is quite significant, which further underscores the importance of introducing the zero-shot CLIP-based classifier.

To further evidence the generality and superiority of our model, we conduct experiments on ten other popular benchmarks across various domains, with the results reported in Fig. 2. As shown, our method surpasses or is on par with the competing baselines on most benchmarks.

[2] For example, on the 16-shot ImageNet, the accuracy is 70.77% when $\boldsymbol{l}$ is set as a learnable vector and 70.42% when constraining all elements in $\boldsymbol{l}$ to have the same value.

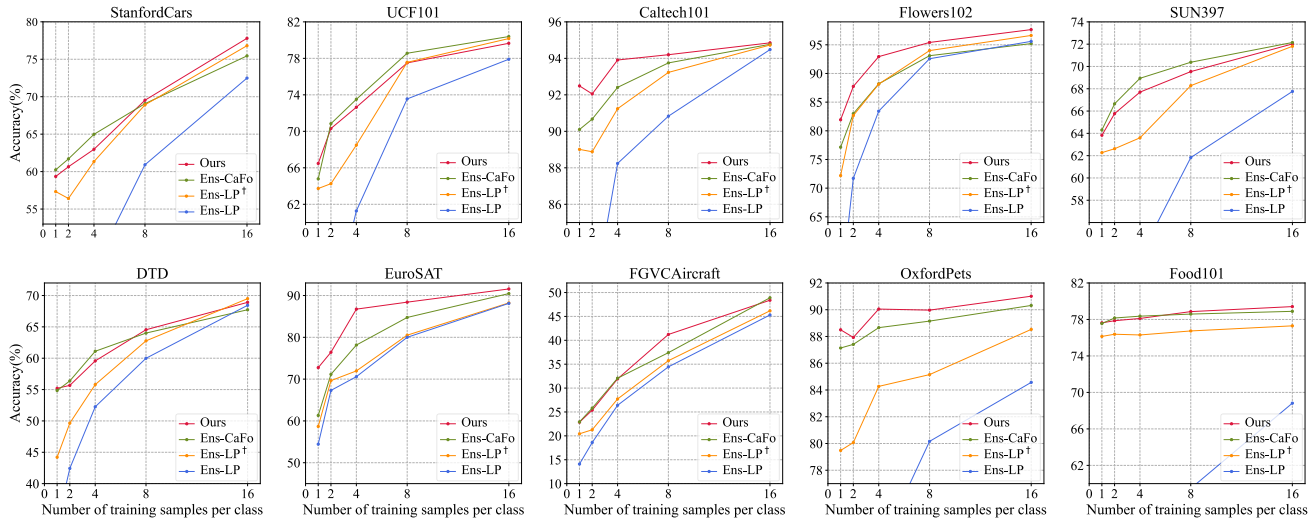


Figure 2. Comparison of low-shot classification accuracy (%) on the ten popular benchmarks.

Shot	1	2	4	8	16
Linear-probe	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
VT-CLIP	60.53	61.29	62.02	62.81	63.92
Tip-Adapter-F	61.32	61.69	62.52	64.00	65.51
CALIP-FS	61.35	62.03	63.13	64.11	65.81
Ours	**63.07**	**65.17**	**67.50**	**69.31**	**70.77**

Table 2. Comparison with leading methods of low-shot classification accuracy (%) on ImageNet.

Datasets	**Source**	**Target**	
	ImageNet	-V2	-Sketch
Ens-CaFo	68.53	59.62	36.12
Ens-LP	66.37	55.08	24.76
Ens-LP†	70.13	59.86	34.66
Ours	**70.77**	**61.30**	**36.58**

Table 3. Test accuracy (%) on OOD datasets.

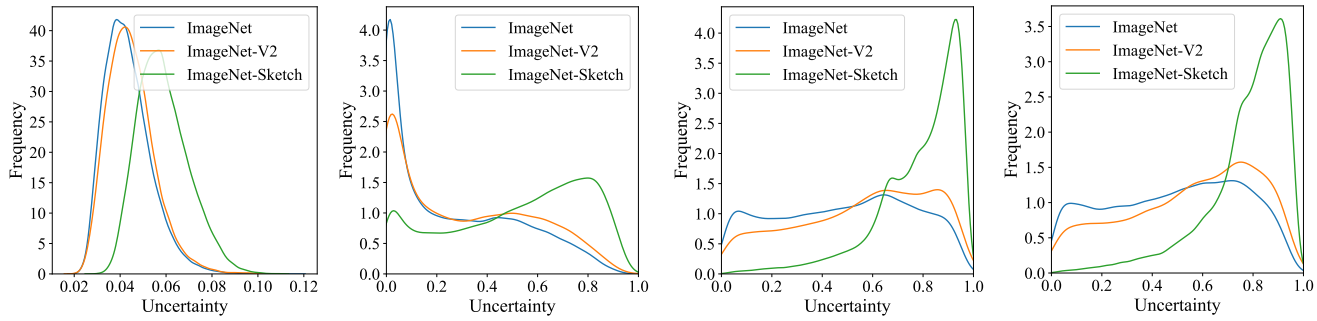
We also compare our method to leading CLIP-based low-shot learners, including CLIP-Adapter [15], Tip-Adapter-F [56], CoOp [59], and CALIP-FS [17] on ImageNet [11]. All these methods use the CLIP model with ResNet-50 architecture, the same as ours. The results in Tab. 2 show that our method consistently achieves higher accuracy than the leading approaches, which indicates the necessity of assembling complementary prior knowledge from various pre-trained models for low-shot classification.

5.3. Evaluation on OOD Data

We next evaluate the robustness and the quality of uncertainty estimates of our method on OOD data.

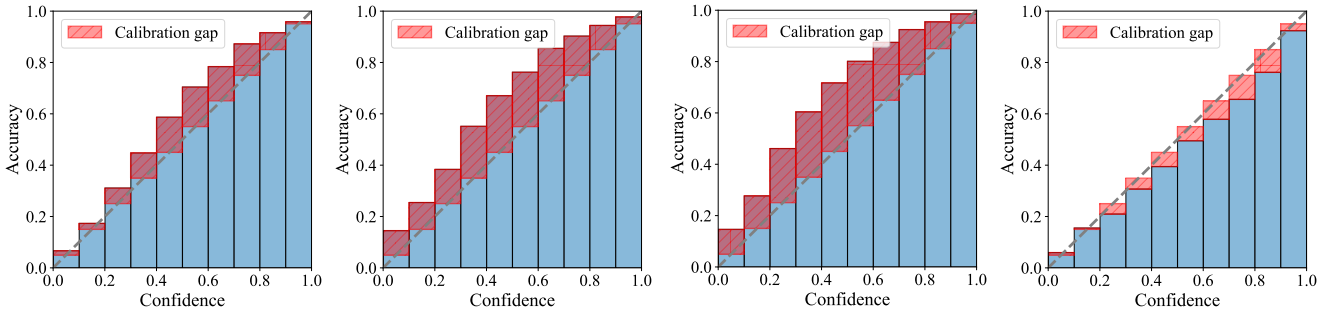
OOD robustness. We use our model trained on 16-shot ImageNet to evaluate OOD samples from ImageNet-V2 [46] and ImageNet-Sketch [22]. ImageNet-v2 is an ImageNet test set collected using the original labeling protocol, with 10 samples per class. ImageNet-Sketch shares the same classes as ImageNet, but all images are sketches. As shown in Tab. 3, our model exhibits superior OOD robustness compared to the ensemble baselines on both ImageNet-V2 and ImageNet-Sketch.

Quality of uncertainty estimates. We then assess the quality of our uncertainty estimates on the above OOD datasets. We collect the predictive uncertainty estimates yielded by our model for both in-distribution data points and OOD ones and depict the histogram in Fig. 3, where the results of the baselines are also included. As the baselines are deterministic, we take one minus the prediction confidence as their uncertainty estimate. As implied by the histograms, our model does not regard ImageNet-V2 as OOD data, which aligns with the fact that the distribution of ImageNet-V2 is as similar as possible to the original ImageNet [43]. On the other hand, our method can clearly identify the OOD ImageNet-Sketch dataset. For the other three baselines, while we can also observe that the differences between ImageNet and ImageNet-V2 are smaller than those between ImageNet and ImageNet-Sketch, the manifestations of these properties are not as pronounced as in our approach. We also provide additional illustrative figures demonstrating the utilization of uncertainty estimates,



(a) Ours (b) Ens-CaFo. (c) Ens-LP. (d) Ens-LP†.

Figure 3. Histogram for uncertainty estimates. We evaluate different ensemble methods on ImageNet, ImageNet-V2, and Imagenet-Sketch.



(a) Ours (b) Ens-LP (c) Ens-LP† (d) Ens-CaFo

Figure 4. Realibility diagrams of the four ensemble methods.

shown in the Appendix.

We further quantitatively estimate the effectiveness of the uncertainty estimates by using them to distinguish ImageNet-Sketch from ImageNet. The AUROCs for our method, Ens-LP, Ens-LP†, and Ens-CaFo to distinguish between ImageNet and ImageNet-Sketch are 0.8545, 0.8249, 0.8253, and 0.7546 respectively. These results align with our previous analyses and validate the superior reliability of our uncertainty estimates.

5.4. Model Calibration

We then evaluate the model calibration of the proposed method by the ECE metric [16]. For our method, we take the maximum element in $\mathbb{E}[\boldsymbol{f}^*]$ as predictive confidence to calculate ECE. The results are presented in Tab. 4, and our model is slightly worse than Ens-CaFo. However, according to [39], ECE leaves ambiguity in both its binning implementation and the calibration computation for multi-class scenarios. Its robust variant, TACE [39], can be a better alternative. As shown in Tab. 4, our method enjoys the best TACE compared to all baselines.

We further present the reliability diagrams of the four methods in Fig. 4. The model calibration is good if the reliability diagram is close to the diagonal. As shown, compared to Ens-LP and Ens-LP†, our method and Ens-CaFo

Method	Ens-LP	Ens-LP†	Ens-CaFo	Ours
ECE	0.1489	0.1858	**0.0577**	0.0786
TACE	0.0462	0.0477	0.0545	**0.0169**

Table 4. ECE and TACE of the four ensemble methods. All experiments are conducted on ImageNet.

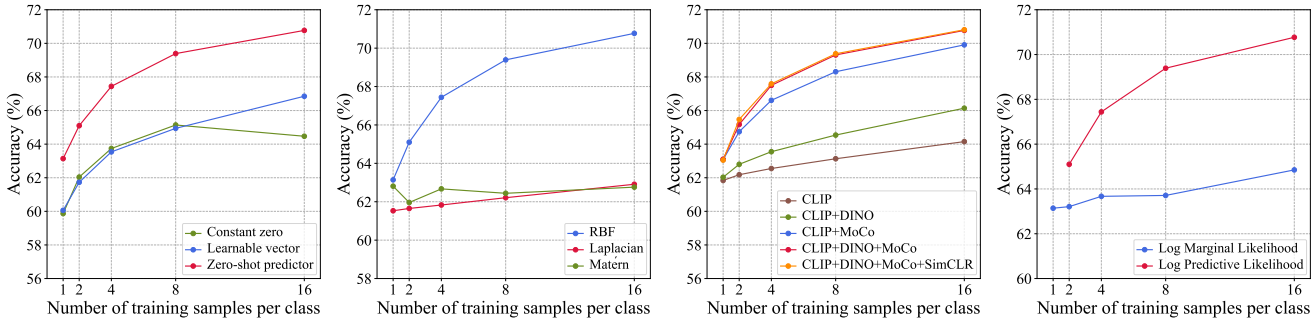
are more well calibrated. Besides, our method, Ens-LP, and Ens-LP† tend to be underconfident, and the Ens-CaFo tends to be overconfident. Combining the results in Tab. 4 and Fig. 4 yields the conclusion that our method enjoys good model calibration.

5.5. Ablation Study

In this section, we offer ablation studies for our method, including an examination of the mean and kernel of the GP, an investigation into how optimization objectives impact the results, and some visualization results.

GP mean. We investigate the impact of the GP mean on the final results in Fig. 5a. As shown, when the mean function equals zero or a learnable vector, prior knowledge is not incorporated into the GP model, and the final few-shot classification performance is unsatisfactory.

GP base kernel. We then delve into the specification



(a) (b) (c) (d)

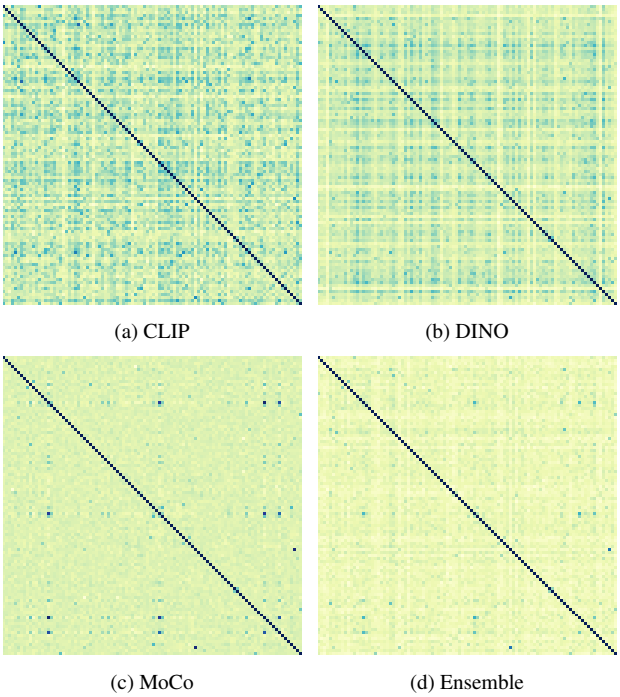
Figure 5. Ablation studies on (a) GP mean, (b) GP base kernel, (c) Pre-trained model, and (d) hyper-parameter optimization objective. All experiments are conducted on ImageNet.

of the GP base kernel. The GP base kernel takes the RBF formula, we also explore other formulas of base kernels, e.g., the Laplacian kernel and Matérn kernel. The results on ImageNet are presented in Fig. 5b. It is evident that the RBF kernel performs best.

Pre-trained model. Further, we test using various pre-trained models to specify the GP kernel. The results on ImageNet are shown in Fig. 5c. It is evident that integrating multiple sources of prior knowledge provided by different pre-trained models leads to substantial advantages. We observe that assembling three pre-trained models already provides comprehensive prior knowledge of ImageNet, and additional integration of pre-trained models like SimCLR [7] does not significantly improve performance.

Objective. As pointed out, using the marginal likelihood can potentially result in underfitting or overfitting [25, 34]. Thus, we advocate the predictive likelihood for tuning hyper-parameters. We perform an empirical study on the objective for hyper-parameter optimization in Fig. 5d. The results clearly echo such an argument and support the use of predictive likelihood for hyper-parameter optimization.

Visualization of the prior kernels. In Fig. 6, we illustrate the data similarities given by the prior deep kernels defined with various pre-trained models. The data points are randomly sampled from ImageNet. The results reflect that distinct prior knowledge regarding data similarities is embedded in these models, and through the kernel ensemble approach, our method can achieve knowledge integration.



(a) CLIP (b) DINO

(c) MoCo (d) Ensemble

Figure 6. Visualization of prior kernel similarities.

6. Conclusion

This work presents a simple and effective Bayesian approach for low-shot image classification. We develop a GP framework to flexibly incorporate diverse prior knowledge from pre-trained models. Extensive experiments showcase the superiority and strong generalization capabilities of our method. More importantly, we demonstrate that the uncertainty given by our method is well-calibrated. Our method will likely enable intriguing applications such as OOD detection by leveraging the uncertainty estimates. Overall, our study demonstrates the exceptional power of Bayesian methods in the large model era and aids in paving the path for future algorithmic improvements in low-shot learning.

Acknowledgments

This work was supported by NSF of China (No. 62306176, 62106121), Natural Science Foundation of Shanghai (No. 23ZR1428700), the Key Research and Development Program of Shandong Province, China (No. 2023CXGC010112), CCF-Baichuan-Ebtech Foundation Model Fund, and the MOE Project of Key Research Institute of Humanities and Social Sciences (22JJD110001).

References

[1] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*. Springer, 2006. 4

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 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*, pages 446–461. Springer, 2014. 5

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1

[4] Roberto Calandra, Jan Peters, Carl Edward Rasmussen, and Marc Peter Deisenroth. Manifold gaussian processes for regression. In *2016 International joint conference on neural networks (IJCNN)*, pages 3338–3345. IEEE, 2016. 2

[5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 1, 2

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 2

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 1, 8

[8] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020. 1

[9] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9640–9649, 2021. 2

[10] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014. 5

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5, 6

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[14] Li Fei-Fei, Rob Fergus, and Pietro Perona. 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*, pages 178–178. IEEE, 2004. 5

[15] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021. 1, 2, 6

[16] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017. 2, 7

[17] Ziyu Guo, Renrui Zhang, Longtian Qiu, Xianzheng Ma, Xupeng Miao, Xuming He, and Bin Cui. Calip: Zero-shot enhancement of clip with parameter-free attention. *arXiv preprint arXiv:2209.14169*, 2022. 2, 6

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[19] Kaiming He, Ross Girshick, and Piotr Dollár. Rethinking imagenet pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4918–4927, 2019. 2

[20] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 1

[21] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. 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, 2019. 5

[22] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15262–15271, 2021. 6

[23] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018. 2

[24] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021. 1, 2

[25] Tianjun Ke, Haoqun Cao, Zenan Ling, and Feng Zhou. Revisiting logistic-softmax likelihood in bayesian meta-learning for few-shot classification. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 3, 8

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[27] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 491–507. Springer, 2020. 1

[28] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013. 5

[29] Malte Kuss. *Gaussian process models for robust regression, classification, and reinforcement learning*. PhD thesis, echnische Universität Darmstadt Darmstadt, Germany, 2006. 3

[30] Jaehoon Lee, Yasaman Bahri, Roman Novak, Samuel S Schoenholz, Jeffrey Pennington, and Jascha Sohl-Dickstein. Deep neural networks as gaussian processes. *arXiv preprint arXiv:1711.00165*, 2017. 2, 3

[31] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2

[32] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 2

[33] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 2

[34] Sanae Lotfi, Pavel Izmailov, Gregory Benton, Micah Goldblum, and Andrew Gordon Wilson. Bayesian model selection, the marginal likelihood, and generalization. In *International Conference on Machine Learning*, pages 14223–14247. PMLR, 2022. 3, 8

[35] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 5

[36] Radford M Neal. Bayesian learning for neural networks, 1996. 2

[37] Hannes Nickisch and Carl Edward Rasmussen. Approximations for binary gaussian process classification. *Journal of Machine Learning Research*, 9(Oct):2035–2078, 2008. 2

[38] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008. 5

[39] Jeremy Nixon, Michael W Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. Measuring calibration in deep learning. In *CVPR workshops*, 2019. 2, 7

[40] Sebastian W Ober, Carl E Rasmussen, and Mark van der Wilk. The promises and pitfalls of deep kernel learning. In *Uncertainty in Artificial Intelligence*, pages 1206–1216. PMLR, 2021. 3

[41] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012. 5

[42] Massimiliano Patacchiola, Jack Turner, Elliot J Crowley, Michael O'Boyle, and Amos J Storkey. Bayesian meta-learning for the few-shot setting via deep kernels. *Advances in Neural Information Processing Systems*, 33:16108–16118, 2020. 3

[43] Longtian Qiu, Renrui Zhang, Ziyu Guo, Ziyao Zeng, Yafeng Li, and Guangnan Zhang. Vt-clip: Enhancing vision-language models with visual-guided texts. *arXiv preprint arXiv:2112.02399*, 2021. 1, 6

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 2, 4

[45] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 5

[46] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR, 2019. 6

[47] Jürgen Schmidhuber. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. PhD thesis, Technische Universität München, 1987. 2

[48] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[49] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 5

[50] Sebastian Thrun and Lorien Pratt. Learning to learn: Introduction and overview. *Learning to learn*, pages 3–17, 1998. 2

[51] Vishaal Udandarao, Ankush Gupta, and Samuel Albanie. Sus-x: Training-free name-only transfer of vision-language models. *arXiv preprint arXiv:2211.16198*, 2022. 2

[52] Christopher Williams and Carl Rasmussen. Gaussian processes for regression. *Advances in neural information processing systems*, 8, 1995. 3

[53] Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006. 1, 2, 4

[54] Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P Xing. Deep kernel learning. In *Artificial intelligence and statistics*, pages 370–378. PMLR, 2016. 2

[55] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010. 5

[56] Renrui Zhang, Rongyao Fang, Wei Zhang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*, 2021. 1, 2, 6

[57] Renrui Zhang, Xiangfei Hu, Bohao Li, Siyuan Huang, Hanqiu Deng, Hongsheng Li, Yu Qiao, and Peng Gao. Prompt, generate, then cache: Cascade of foundation models makes strong few-shot learners. *arXiv preprint arXiv:2303.02151*, 2023. 1, 2, 5

[58] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16825, 2022. 1

[59] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 1, 2, 6