Dual Visual Prompting with Context-Modulated Diffusion Prompts

Huan Wang, *Student Member, IEEE,* Haoran Li, *Student Member, IEEE,* Huaming Chen, *Member, IEEE,* Jun Yan, Jiahua Shi, and Jun Shen✉, *Senior Member, IEEE*

***Abstract*—Prompt learning has emerged as an efficient tuning paradigm for fine-tuning powerful pre-trained models on downstream tasks in specific domains. Existing efforts mainly focus on dataset-level implicit embeddings by introducing extra learnable parameters instead of fully fine-tuning large-scale visual models. However, we find that these static post-training prompts are not flexible enough to adapt various input instances within the same dataset, which might lead to the loss of the model's generalization capability. To leverage the meaningful contextual information of each input instance, in this paper, we propose a straightforward yet effective method, termed CoMoDP, to enhance visual prompt learning with *Co*ntext-*Mo*dulated *D*iffusion *P*rompts. Specifically, CoMoDP is a dual-visual prompting scheme that comprises two key components: *(i) a unified visual prompt designer*, producing dataset-level implicit embedding as unified prompts for efficient adaptation without corrupting the underlying information of the original image; and *(ii) a diffusion prompt simulator*, leveraging diffusion model's meticulous understanding of semantic structure and texture edges in the images to dynamically generate instance-level implicit embedding as diffusion prompts for input samples. Moreover, to reduce the overfitting of prompts, we also introduce momentum alignment, a self-regulating strategy that restricts the optimization region of prompts in both feature and logit spaces. Extensive experiments on various standard and few-shot datasets demonstrate that our method brings substantial improvements and yields strong domain generalization performance, compared to the state-of-the-art methods. We also demonstrate both zero-shot and out-of-distribution performance to establish the utility of our dual-visual prompting scheme CoMoDP and the efficiency of each component, without involving excessive parameters.**

***Index Terms*—Parameter-Efficient Fine-Tuning, Visual Prompt Learning, Vision-Language Learning, Diffusion Models.**

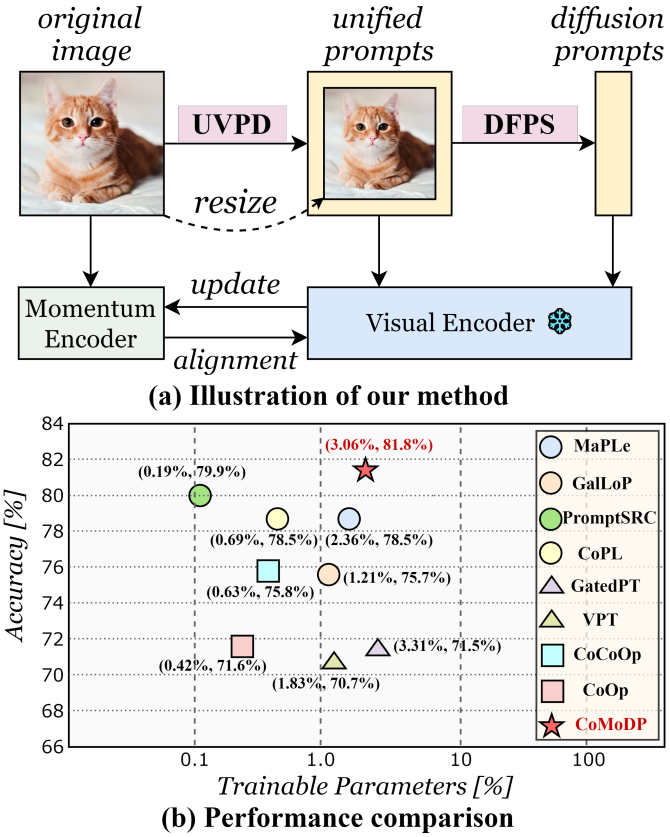

Fig. 1. **(a)** The core idea of the proposed CoMoDP with two key modules: UVPD and DFPS. **(b)** Comparison of our CoMoDP with other prompt-based baselines, showing the average HM accuracy over 11 standard datasets (details in Tab. II). Our proposed CoMoDP yields solid improvement and achieves a better trade-off between performance and trainable parameters usage against the state-of-the-art competitors.

I. Introduction

Currently, the rise of powerful pre-trained models such as ViT [1] and CLIP [2] has demonstrated striking generalization capabilities for various downstream tasks, especially for zero-shot visual perception [3]–[5]. These vision or vision-language models are trained on large-scale datasets, which allow them to learn an influential feature embedding that exhibits significant performance improvements in downstream tasks. Furthermore, by sharing the embedding space, these models align representations without biasing them toward labels containing specific downstream task information [6], [7].

To better optimize and fine-tune the pre-trained models for various domain-specific downstream tasks, parameter-efficient prompt-based transfer paradigm has been introduced [8]–[12], which aims to adapt the well-learned representation while keeping the model's parameters frozen. Especially, prompt learning [13], [14] introduces a few learnable prompt tokens or vision perturbations to regulate the pre-trained models for downstream tasks, such as the conventional methods VPT [15] and VP [16]. However, since building prompt tokens with rich semantics requires domain-specific knowledge and appropriate prompt templates, conventional prompt learning is not scalable. The CoOp [9] was inspired by prompt learning in natural language processing [13], [17], [18], utilizing learnable vectors to model the prompt tokens and preserve semantic relationships. However, the learned context by CoOp is not generalizable to wider unseen classes within the same dataset. To address the generalization problem, CoCoOp [11] was then

H. Wang, H. Li, J. Yan, and J. Shen✉ (Corresponding Author) are with the School of Computing and Information Technology, University of Wollongong, Wollongong, Australia. (E-mail: {hw226, hl644}@uowmail.edu.au, {jyan, jshen}@uow.edu.au).

H. Chen is with the School of Electrical & Computer Engineering, University of Sydney, Sydney, Australia (E-mail: huaming.chen@sydney.edu.au).

H. Li and J. Shi are with the Centre for Nutrition and Food Sciences, University of Queensland, Brisbane, Australia (E-mail: jiahua.shi@uq.edu.au).

proposed to dynamically regulate the prompt tokens based on the image context. CoCoOp generates conditional vectors for each image, applying different weights to all prompt tokens, in order to reduce sensitivity for class shift. The more recent studies find that the CoOp-based methods would lead to prompt overfitting and knowledge forgetting, and gradually ignore essential general linguistic knowledge [19], [20]. Although these statically designed prompts are optimized with respect to all data samples of the same dataset, the generalizable context of each input instance, which can potentially serve as effective guidance for visual prompt learning, is overlooked.

Recently, the development of diffusion models (DMs) [21]–[24] has provided new opportunities. The large-scale text-to-image diffusion models have demonstrated remarkable capabilities in generating diverse and high-resolution images, which shows strength in capturing intricate details and patterns from underlying data. Among these models, stable diffusion (SD) [23] exhibit superior performance in generating high-quality images with rich texture, diverse content, and reasonable structures. Besides, SD models utilize cross-attention mechanisms to align textual prompts with corresponding image regions during the denoising process. The integration of cross-attention mechanisms enables SD models to represent and generate complex visual concepts described in linguistic texts, which demonstrates their potential ability to understand both high-level and low-level visual concepts regarding what an image contains. Moreover, SD models have recently been testified as effective classifiers [25], [26] and zero-shot semantic correspondence learners [27], [28]. These SD models can weave from scratch with rich and innovative depictions of high fidelity, and further capture underlying properties, positions, and correspondences of objects that make up the resulting generated image, along the whole process. Such compelling semantic generation, combination, and content-understanding capabilities of the diffusion models motivate us to speculate: ***is it possible to extract the semantic perception knowledge about the visual concepts inherent in pre-trained diffusion models as dynamic guidance on each input image in visual prompt learning?*** However, utilizing the SD models for visual prompt learning remains challenging. The SD models generate high-fidelity images by initiating from random noise and iteratively refining it through a learned denoising process conditioned on textual prompts. While producing visual prompts from images involves approximating feature distillation, aiming to infer underlying patterns that could have led to the observed image. Although rich internal representations are learned in the pre-training process of SD models from diverse and multi-scale image-text pairs, it is still unclear how to extract this knowledge and whether it can benefit visual prompt learning.

In this paper, we explore how to leverage the meaningful internal representations learned by the pre-trained diffusion models to dynamically optimize each instance in visual prompting. We introduce a novel dual-visual prompting paradigm to enhance dynamic perception with **Co**ntext-**Mo**dulated **D**iffusion **P**rompts in visual prompt learning, termed CoMoDP. The key novelty of CoMoDP is to construct dataset-level unified prompts and instance-level diffusion prompts, providing task-specific domain knowledge and sample-specific dynamic guidance for visual prompting, rather than fixed static prompts to adapt all data samples. As illustrated in Fig. 1 (a), CoMoDP consists of two critical components: a *Unified Visual Prompt Designer* (UVPD) and a *Diffusion Prompt Simulator* (DFPS). **First**, UVPD shrinks the original image into a smaller size and then harmonizes unified prompts through different strategies (see Fig. 3), aiming to capture task-specific domain knowledge via unified prompts at the pixel space. Unlike directly adding extra pixels [16], UVPD module avoids destroying the underlying visual representations of the original image, via a shrink-and-harmonize approach. **Second**, DFPS leverages the fine-grained visual perception capabilities of diffusion models to simulate diffusion prompts, offering sample-specific dynamic guidance for each input instance. The diffusion prompts can dynamically provide more distinguishing guidance for input samples that are more semantically relevant and contextually meaningful through the diffusion model's meticulous understanding about visual concepts (*e.g.*, structures, textures, edges) in an image. To achieve this, we further learn a lightweight *Context Modulator Network* (CM-Net) to project fused semantic features extracted from the pre-trained diffusion model's UNet into the diffusion prompt embeddings, which are then injected into the vision encoder as conditioning signals. Additionally, to reduce the overfitting of prompts, we introduce momentum alignment, a self-regulating strategy that restricts the optimization region of prompts in both feature and logit spaces. Benefiting from such a dual-visual prompting scheme, CoMoDP exhibits both efficient and effective performance across multiple downstream datasets, as shown in Fig. 1 (b). Our main contributions are:

- We focus on exploiting the fine-grained visual representation capabilities of the pre-trained diffusion model to facilitate the dynamic perception of the visual encoder for each input instance in visual prompt learning.
- We introduce a dual-visual prompting scheme, termed as CoMoDP, capturing task-specific domain knowledge and sample-specific guidance signals to promote the model's generalizability by two complementary modules: UVPD produces dataset-level implicit embedding as the unified prompts for efficient adaptation at the pixel space; DFPS generates instance-level implicit embedding as diffusion prompts to dynamically characterize class shift for each input, more robust to samples with unseen classes. These two components together enable CoMoDP to learn more generalized contexts in visual prompt learning.
- We conduct extensive experiments with CoMoDP under a variety of image-classification settings and demonstrate substantial performance improvements, especially in various zero-shot and few-shot data scenarios. Accompanied with a series of promising results and ablation studies, it validates the effectiveness of our proposed approach.

II. Related Work

A. Vision Language Models

Vision-language models have shown great potential in learning a joint embedding space to obtain generic representations

by aligning images and texts [2], [9], [29], [30]. The typical vision-language model consists of three key elements: visual encoder, text encoder, and the design of loss functions [2], [31]. Specially, we mainly review studies focused on how to align the visual and linguistic contents to facilitate representation learning, which includes primarily two aspects: (i) optimizing the image and text encoder separately; and (ii) jointly modeling image and text modalities. The former focuses on independently designing and learning modules for processing images and texts, and aligning their outputs by an additional interaction module [32], [33]. The images are often encoded using neural network architectures [1], [34], and the texts are encoded using large language models or pre-trained sequence models [35], [36]. One of the biggest milestones in multi-modal research is the CLIP model [2], which use a dual-encoder-based model and matched image and text pairs during training. ViLBERT [37] extends the popular BERT [35] architecture to a multi-modal two-stream model, processing both visual and textual inputs in separate streams. ViLT [38] drastically simplifies the processing of visual inputs in a convolution-free manner to improve efficiency and enhance expressiveness. The other direction aims to learn joint representations of both the image and text modalities using deep learning architecture [39]–[41]. ALBEF [39] introduces a contrastive loss to align the image and text representations before fusing them through cross-modal attention to boost model performance. ALPro [42] devises a video-text contrastive loss to align unimodal video-text features at the instance level, which eases the modeling of cross-modal interactions. The main motivation of these works is to learn generic representations using image and text supervision and regularize them in the same embedding space.

B. Visual Prompt Learning

Prompt learning [13], [14], as a popular and efficient tuning paradigm, originally appeared in natural language processing (NLP) for adapting the pre-trained language models to various downstream datasets and tasks [14], [43]. These prompts can be handcrafted for the specific task-relevant datasets [44] or learned automatically via the gradient backpropagation while keeping the model parameters fixed [9]. Considering the efficiency in terms of parameter size and convergence rate, prompt learning is rapidly expanded to adapt large-scale vision foundational models [45]–[47]. Further, CoOp [9] and CoCoOp [11] fine-tuned the pre-trained CLIP model by optimizing a continuous set of prompt vectors in multi-modal scenarios. CoCoOp [11] was built on top of CoOp to improve the model performance through conditioning on each image input to enhance generalizability. Meanwhile, VP [16] and VPT [15] introduced prompt learning in the vision branch, imposing visual prompt tuning at embedding or pixel level by learning the tokens. Despite the pivoting successes of these prompt-based methods, they are mostly static post-training, which is not flexible enough to adapt all samples in the generalized scenarios. In contrast, our method enhances visual prompting with context-modulated diffusion prompts to conditionally optimize each input.

C. Diffusion Representations

Recently, diffusion models have shown significant advances in image generation, editing, and stylization [23], [24], suggesting that they contain rich internal representations to be exploited for various downstream tasks. DIFT [27] extracts diffusion features to establish correspondence that emerges among images in image diffusion models without any explicit supervision. Plug-and-Play [48] injects the extracted features from a single layer of the diffusion UNet to preserve image structure in text-guided editing. SODA [49] distills a source image into a compact representation, which in turn guides the generation of related views to capture visual semantics. Diffusion Hyperfeatures [50] further consolidates multi-scale and multi-timestep features into feature descriptors to identify and feed the most relevant features for the specific task. While these works have analyzed the underlying diffusion representations and proposed using them for downstream tasks, how to extract more expressive diffusion features still remains unclear [51], [52]. In this paper, we focus on distilling the most semantically relevant diffusion features that take the collection of a source image view from the diffusion process as inputs, and produce corresponding diffusion prompts as outputs.

III. Preliminaries and Background

A. Pre-trained Models

Contrastive Language-Image Pre-training (**CLIP**) model [2] is a large-scale pre-trained multi-modality model derived from 400 million image-text pairs by an image encoder $f_i(\cdot)$, and a text encoder $f_t(\cdot)$. CLIP performs zero-shot classification based on the similarity between image feature $f_i(x)$ and text feature $f_t(\mathtt{T}_{y_x})$. The text feature $f_t(\mathtt{T}_{y_x})$ is obtained through the combination of the corresponding label y_x and the text caption $\mathtt{T}_{y_x}$ as '*a photo of a* $\mathtt{Z}_{y_x}$', where $\mathtt{Z}_{y_x}$ denotes the class name of the label y_x (*e.g.*, $\mathtt{Z}_{y_x} \rightarrow$ 'cat'). Given the image x and text captions, CLIP outputs a prediction as $\mathtt{y}_{\mathtt{pred}}$:

$$\mathtt{y}_{\mathtt{pred}} = \arg\max_{y_c}(cos(f_t(\mathtt{T}_{y_c}) \cdot f_i(x))), \tag{1}$$

where $\mathtt{T}_{y_c}$ is the text caption of label y_c, and $cos(\cdot)$ denotes the cosine similarity. Benefiting from large-scale pre-training and alignment of images and text, CLIP is robust for various visual appearance changes [53], [54]. Vision Transformer (**ViT**) [1] is a transformer-based model for various visual-related tasks with generalization and adversarial robustness. Given the image $x \in \mathbb{R}^{H\times W\times C}$, ViT first reshapes x into the flattened patches as $x_p \in \mathbb{R}^{N\times P^2\times C}$, where P is the resolution of the image patch and $N = H\times W/P^2$ is the number of image patches. Then, the image embeddings are input to the ViT transformer encoder (including several attention and MLP blocks). ViT considers last MLP layer's vectors, which are related to the special class token embedding, as predictions, and ignores the other outputs.

B. Visual Prompt Tuning

Prompt tuning adapts frozen pre-trained models to various tasks by introducing tunable parameters to the input space. Visual Prompt Tuning (**VPT**) [15] introduces the visual

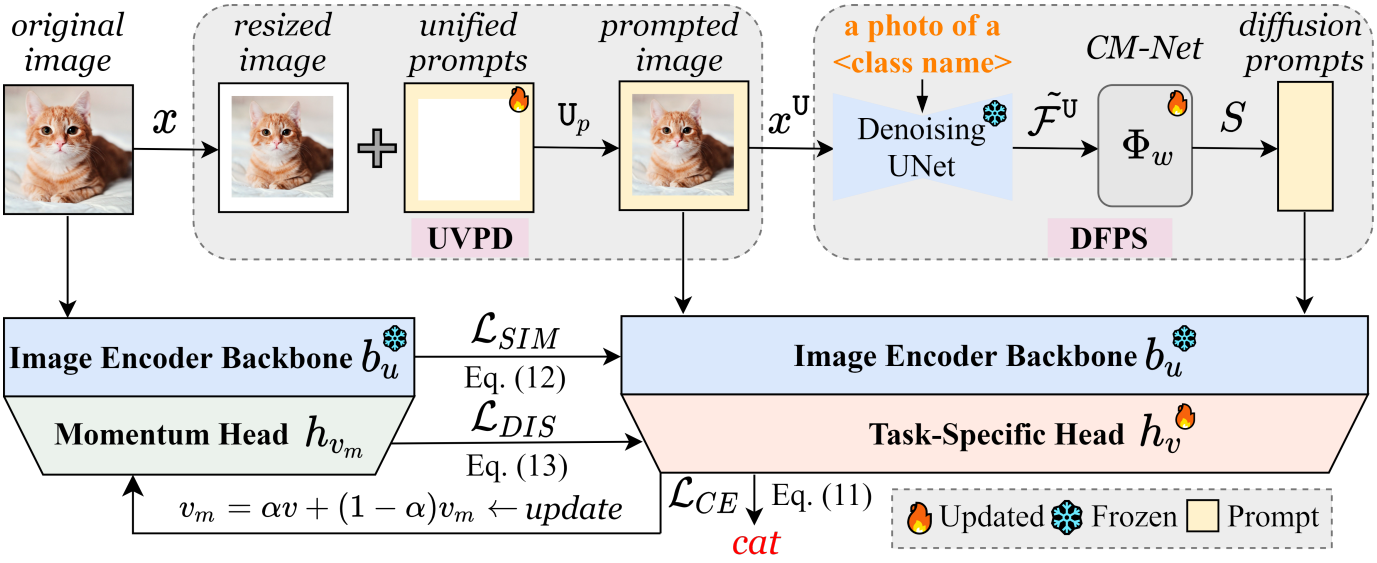

Fig. 2. **Overview of our proposed CoMoDP**. The UVPD module optimizes the unified prompts $\mathtt{U}_p$ to capture task-specific domain knowledge (Sec. IV-A). The DFPS module generates the diffusion prompts S to provide sample-specific semantic guidance (Sec. IV-B). The pseudo-code is given in Algorithm 1.

prompts at the embedding level, which are divided into VPT-Shallow and VPT-Deep according to the insertion position. Among them, VPT-Shallow inserts prompts into the first Transformer layer formulated as $L_1(x_0, P_0^v, E_0)$, where x_0 is the `[cls]` token, P_0^v is the learnable prompts and E_0 is the image patch embeddings. Similarly, VPT-Deep is formulated as $L_i(x_{i-1}, P_{i-1}^v, E_{i-1})$ with prompts being added at every layer. Another typical visual prompt-based approach is Visual Prompting (**VP**) [16] at the pixel level, which learns a single image perturbation to steer visual models. The goal of VP is to adapt pre-trained models by only modifying the pixels of the input image x to form a prompted image $x + v_\phi$ by maximizing the likelihood $P(y|x+v_\phi)$, typically implemented via minimizing the cross-entropy loss on perturbed inputs, where y is the corresponding label of x. VP explores the power of learnable prompts at the pixel space. However, VP's performance appears to be less promising due to directly adding together the prompts and the image, which might corrupt the image's underlying information and thus limit the optimization of prompts. On the other hand, these prompts are added to all the samples within the same dataset, resulting in an equal weighting for each input, which then leads to the loss of the model's original generalization capability.

C. Diffusion Models

The diffusion model learns to transform noisy data (typically for a Gaussian distribution) to the high-quality samples of the desired data distribution. Given the image sample x from an underlying distribution $p(x)$, the forward diffusion process is formulated as a Markov chain that incrementally adds random Gaussian noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ to the image x_0, the x_t is the noise image after adding t-steps noise:

$$x_t = \sqrt{1-\gamma_t} x_{t-1} + \sqrt{\gamma_t}\epsilon_t, \ t \in \{1, ..., T\}, \tag{2}$$

where T is the number of diffusion timesteps, $\gamma_t \in (0, 1)$ is an adjustable time variance schedule [21]. By leveraging the additive property of the Gaussian distribution, Eq. (2) can be further reformulated as $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t$, where $\alpha_t = 1 - \gamma_t$ and $\bar{\alpha}_t \doteq \prod_{i=1}^{t} \alpha_i$. On this basis, the image x_0 can be obtained from a random noise $x_t \sim \mathcal{N}(0, \mathbf{I})$ by reversing the above forward diffusion process [21]:

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)), \ t \in \{T, ..., 1\}, \tag{3}$$

where ϵ_θ denotes the denoiser network with the parameters θ to predict noise ϵ, and the denoiser is usually implemented as a UNet architecture [55]. Further, the denoiser ϵ_θ is optimized using the loss $\mathcal{T}_\theta$ as follows:

$$\mathcal{T}_\theta = \mathbb{E}_{\epsilon, x_0, t}[||\epsilon_\theta(x_t, t) - \epsilon||_2^2], \ \epsilon \sim \mathcal{N}(0, \mathbf{I}), \tag{4}$$

where ϵ is the noise in the forward diffusion process. Besides, the diffusion model's loss in Eq. (4) can be easily extended to a conditional diffusion paradigm by adding a condition $\mathbf{c}$ to the ϵ_θ. Thus, the training objective should be modified as:

$$\mathcal{T}_\theta = \mathbb{E}_{\epsilon, x_0, t, \mathbf{c}}[||\epsilon_\theta(x_t, t, \mathbf{c}) - \epsilon||_2^2], \ \epsilon \sim \mathcal{N}(0, \mathbf{I}). \tag{5}$$

IV. Methodology

CoMoDP leverages UVPD and DFPS modules to provide unified domain knowledge and explicit semantic guidance for visual prompting while reducing the overfitting of prompts via momentum alignment regulating. An overview of the proposed CoMoDP framework is shown in Fig. 2.

A. Unified Visual Prompt Designer

To capture dataset-level domain knowledge in visual prompt learning, we speculate that a well-generalizable global prompt should capture domain-level structure without encoding overly specific class-discriminative patterns, thereby supporting generalization across classes. Meanwhile, it should optimize this prompt independently to avoid corrupting the underlying representations of the original sample. Hence, we introduce a

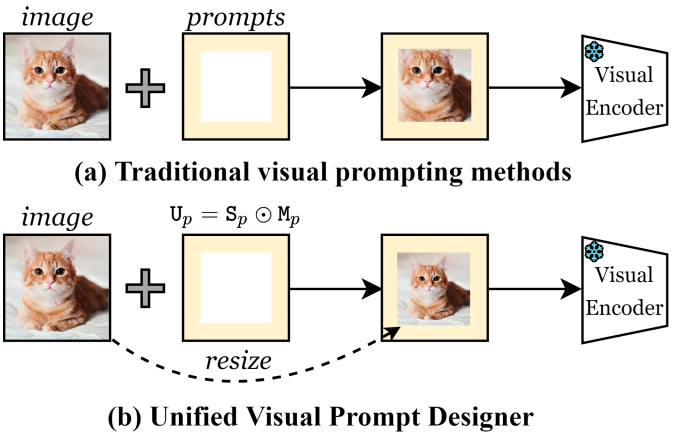

Fig. 3. **(a)** Traditional visual prompting methods (*e.g.*, VP [16]) *may* ***corrupt*** *the underlying information of the original image and* ***focus less*** *on the local discriminative regions in the image* (*e.g.*, the cat ears are heavily obscured by the prompts). **(b)** Illustration of UVPD module, with a shrink-and-harmonize way: resizing the original image and extending the unified prompts.

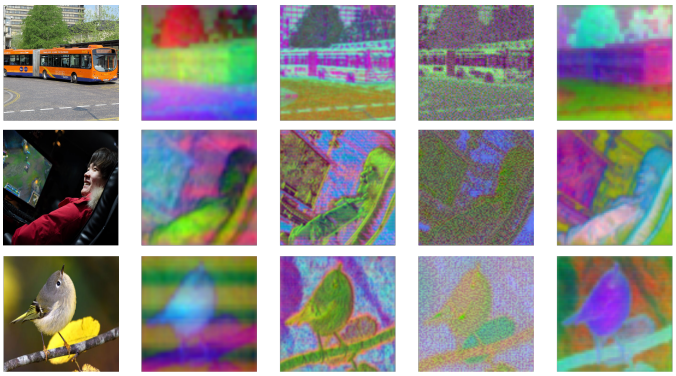

Fig. 4. The PCA visualization of the early (layer 2), later (layer 4), and fused features (see Eq. (9)) in the diffusion model, where the first three components of PCA-computed serve as color channels for each image.

Unified Visual Prompt Designer (UVPD) module to fully exploit the potential of visual prompting at the pixel space, in a shrink-and-harmonize way, from the perspectives of uniformity and transferability. We observe that the previous work [16] directly add prompts to the image, which might corrupt the meaningful information from the original image (*e.g.*, prompts may be considered as noise, reducing the focus on the discriminative foreground image). An intuitive solution is to keep prompts and the image non-overlapping, *i.e.*, to optimize the prompts as an extra and independent learnable element (see Fig. 3).

Specifically, our UVPD module optimizes task-specific image perturbations as unified prompts: (i) *Shrink*: the image is resized to a smaller size, maintaining integrity; (ii) *Harmonize*: the unified prompts are padded around the resized image. In practice, we explore different harmonized templates and find that evenly distributing the unified prompts around the resized image achieves the best performance (*i.e.*, Fig. 3 (b)). We also analyze the effects of different prompt sizes and harmonized templates in Tab. VI. Given a frozen pre-trained vision encoder

TABLE I
SUMMARY OF VARIABLE DEFINITIONS AND DIMENSIONS IN SEC. IV.

Variable	Definition	Formalization
m	original image resolution	default as 224
p	prompt padding size	default as 30
3	RGB channels of the image	channels
$\mathtt{S}_p$	prompt parameters	$\mathbb{R}^{m\times m\times 3}$
$\mathtt{M}_p$	masked parameters	$\mathbb{R}^{m\times m\times 3}$
$\mathtt{U}_p$	unified prompts	$\mathbb{R}^{m\times m\times 3}$
$\odot$	element-wise multiplication	operator
$+$	element-wise addition	operator
$\mathtt{Z}_y$	the class name of the label y	text-type
$\mathtt{T}_y$	the text template of the label y	text-type
$\tilde{\mathcal{F}}$	the fused diffusion features	default as $\mathbb{R}^{512}$
S	diffusion prompts	default as $\mathbb{R}^{512}$

F_δ and a downstream task dataset $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$ where x_i is the sample, $y_i \in \{1, ..., \mathcal{C}\}$ is the label, and n denotes the dataset scale. First, the image sample x_i with size m is resized into a smaller size $m - 2 * p$, as the resized image $\hat{x_i}$. Then, we define the unified prompts as $\mathtt{U}_p = \mathtt{S}_p \odot \mathtt{M}_p$, where $\mathtt{S}_p \in \mathbb{R}^{m\times m\times 3}$ is prompt parameters (3 means the RGB channels of the input image, m represents the original image resolution), $\mathtt{M}_p \in \mathbb{R}^{m\times m\times 3}$ is masked parameters to learn the mutual spatial relations of the prompts where the center $(m-2p)^2$ is zeroed (set to zero), p denotes the padding size (pixels), not embedding dimension. We also present Tab. I summarizing variable definitions and dimensions. The unified prompts $\mathtt{U}_p$ are extended to the resized image $\hat{x_i}$ to get a prompted image $x_i^{\mathtt{U}}$ by:

$$x_i^{\mathtt{U}} = \mathtt{U}_p + \hat{x_i}, \text{ where } \mathtt{U}_p = \mathtt{S}_p \odot \mathtt{M}_p, \ \hat{x_i} \overset{resize}{\longleftarrow} x_i, \quad (6)$$

where $x_i^{\mathtt{U}}$ is the prompted image with the size m, the operator $\odot$ denotes the element-wise multiplication to mask prompt parameters, the operator $+$ denotes the element-wise addition at the pixel level. The $\mathtt{M}_p$ potentially encodes priority relationships between unified prompts to enhance the expressiveness. Inspired by gradient statistical normalization [56], [57], we then use the gradient L_2-norm to optimize $\mathtt{U}_p$ by:

$$\mathtt{U}_p \longleftarrow \mathtt{U}_p - \eta \times \left(\frac{\nabla_{\mathtt{U}_p}\mathcal{L}}{||\nabla_{\mathtt{U}_p}\mathcal{L}||_2}\right), \quad (7)$$

where η is the learning rate, ∇ denotes the gradient, and $\mathcal{L}$ is the loss function of CoMoDP (in Eq. (14)). The UVPD module regulates the model by imposing unified prompts, thus learning the most relevant global domain knowledge for the current task's dataset within the uniform pixel space.

B. Diffusion Prompt Simulator

Although the unified prompts $\mathtt{U}_p$ by the UVPD module consider the task-specific domain knowledge for visual prompting, it cannot be dynamically optimized for each input instance in the same dataset. To improve adaptability and generalizability, we further introduce a *Diffusion Prompt Simulator* (DFPS) module. Inspired by the generative and understanding abilities of stable diffusion models, diffusion features (*i.e.*, intermediate feature maps in the diffusion process produced by multiple applying a UNet of the SD model) might have a strong sense of

spatial and generate smooth correspondences about semantic objects. Given an image-pair (x_i, y_i), $y_i \in \{1, ..., \mathcal{C}\}$ is the label, and we let $\mathtt{Z}$ represents the set of all class names, so we can use $\mathtt{Z}_{y_i}$ to denote the class name of the label y_i. In our implementation, the image x_i is fed into the encoder $\boldsymbol{\varepsilon}$ of the SD model to obtain the latent vector $\boldsymbol{\varepsilon}(x_i)$. Then, the vector is further combined with its relevant text prompt $\mathtt{T}_{y_i}$ ='*a photo of a* $\mathtt{Z}_{y_i}$' and sent to the UNet decoder ϵ_θ of the SD model to produce the diffusion features $\mathcal{F}_i$. Note that here we set timestep $t = 0$ of the diffusion process such that no noise is added to the latent vector. Typically, the image x_i is 224×224 and $\mathcal{F}_i$ has 4 feature maps, where the L-th layer's feature map $\mathcal{F}_{i(L)}$ has the spatial size $\{7, 14, 28, 28\}$ ($L \in \{1, 2, 3, 4\}$), and the channel size as $\{1280, 1280, 640, 320\}$. These dimensions correspond to fixed layers in the SD version 1.5 UNet architecture with input image resolution 224×224. Adaptation to other diffusion backbones may require re-deriving these parameters. The diffusion features $\mathcal{F}_i$ can be computed as follows:

$$\mathcal{F}_i = \epsilon_\theta(\boldsymbol{\varepsilon}(x_i), \tau_i, t = 0), \text{ where } \tau_i \leftarrow \boldsymbol{\varphi}(\mathtt{T}_{y_i}), \tag{8}$$

where $\boldsymbol{\varphi}$ denotes the text encoder of the SD model and τ_i is the text feature with the text prompt $\mathtt{T}_{y_i}$. To examine properties, we perform principal component analysis (PCA) on diffusion features $\mathcal{F}_i$ at different layers. As shown in Fig. 4, former layers tend to focus more on semantics and structures, and the latter layers tend to contain detailed information about textures and appearances. To capture both semantics and details, an intuitive idea is to concatenate features from all layers, but this might lead to an unnecessarily high dimension ($1280 + 640 + 320 = 2240$, since the spatial resolution is too low, we do not consider $L = 1$). To reduce the high dimension, we apply PCA for each layer $L \in \{2, 3, 4\}$ and aggregate them to the same resolution, then we connect these PCA-processed features to obtain fused features $\widetilde{\mathcal{F}}_i$:

$$\widetilde{\mathcal{F}}_i = \mathtt{Concat}(\widetilde{\mathcal{F}}_{i(L)}), \ \widetilde{\mathcal{F}}_{i(L)} = \mathrm{PCA}(\mathcal{F}_{i(L)}), \tag{9}$$

where $L \in \{2, 3, 4\}$, we perform PCA as: (i) for $\mathcal{F}_{i(2)}$, we first upsample to the same resolution and apply PCA $\mathbb{R}^{1280 \to 256}$; (ii) for $\mathcal{F}_{i(3)}$ and $\mathcal{F}_{i(4)}$, we also use PCA $\mathbb{R}^{640 \to 128}$ and PCA $\mathbb{R}^{320 \to 128}$. We apply PCA to each diffusion layer's feature map to reduce its channel dimension to a fixed value (*e.g.*, 256), enabling concatenation while preserving the relevant semantic content. Then, we combine them to form fused features $\widetilde{\mathcal{F}}_i$, as shown in the last column of Fig. 4, fused features strike a balance between high-level and low-level visual concepts, focusing on both semantics and textures in the image (*e.g.*, face and computer borders in the second row of Fig. 4). We compare layer's features and fused features in Tab. VII.

We can directly exploit the fused features of the diffusion model as a context-modulated guidance for each input instance in visual prompting. Based on Eq. (8) and Eq. (9), we get the fused features $\widetilde{\mathcal{F}}_i^{\mathtt{U}}$ of the prompted image $x_i^{\mathtt{U}}$. To align the embedding space of the diffusion model with downstream tasks, we learn a lightweight *Context Modulator Network* (CM-Net) Φ_w (Linear-ReLU-Linear) to generate the conditional diffusion prompts $S_i = \Phi_w(\widetilde{\mathcal{F}}_i^{\mathtt{U}})$ for each input $x_i^{\mathtt{U}}$. Note that we only infer via Φ_w to get S_i once. S_i has the shape of $1 \times dim$, where dim is the embedding dimension. Then, S_i is inserted into the first layer L_1 only of the pre-trained model F_δ:

$$\begin{aligned} &[\mathrm{c}_1, \mathrm{Z}_1, \mathrm{E}_1] = L_1([\mathrm{c}_0, S_i = \Phi_w(\widetilde{\mathcal{F}}_i^{\mathtt{U}}), \mathrm{E}_0]), \\ &[\mathrm{c}_j, \mathrm{Z}_j, \mathrm{E}_j] = L_j([\mathrm{c}_{j-1}, \mathrm{Z}_{j-1}, \mathrm{E}_{j-1}]), \ j \in \{2, 3, .., N\}, \end{aligned} \tag{10}$$

where $\mathrm{Z}_j \in \mathbb{R}^{1 \times dim}$ represents the features computed by the j-th layer in the model F_δ, E_0 means the patch embeddings of the image $x_i^{\mathtt{U}}$, N is the number of layers, and c_0 is the `[cls]` token, which produces $y = Head(\mathrm{c}_N)$ as the final output of image representations by the model F_δ. Combined with the UVPD and DFPS modules, we build the CrossEntropy [58] loss and use the model prediction with signal y_i:

$$\mathcal{L}_{CE} = -\mathbf{1}_{y_i} \log(\sigma(F_\delta(x_i^{\mathtt{U}}, \Phi_w(\widetilde{\mathcal{F}}_i^{\mathtt{U}})))), \tag{11}$$

where σ denotes the softmax function. Based on Eq. (11), we maintain the discriminative ability for the original input x_i.

Algorithm 1 PSEUDOCODE OF COMOMDP

TRAINING PROCESS OF COMOMDP

for each training epoch **do**
  for each dataset loader $\mathcal{D}_{batch}$ from $\mathcal{D}$ **do**
    $x, y \leftarrow \mathcal{D}_{batch}$ (get samples and labels)
    // Unified Visual Prompt Designer
    $\hat{x} \leftarrow x$ (shrink original images)
    $x^{\mathtt{U}} \leftarrow \mathtt{U}_p + \hat{x}$ (get prompted images by Eq. (6))
    // Diffusion Prompt Simulator
    $\mathtt{T}_y \leftarrow$ 'a photo of a $\mathtt{Z}_y$' (get text prompts)
    $\tau \leftarrow \boldsymbol{\varphi}(\mathtt{T}_y)$ (get text features)
    $\mathcal{F}^{\mathtt{U}} \leftarrow \epsilon_\theta(\boldsymbol{\varepsilon}(x^{\mathtt{U}}), \tau, t = 0)$ (get diffusion features)
    $\widetilde{\mathcal{F}}^{\mathtt{U}} \leftarrow \mathcal{F}^{\mathtt{U}}$ (get fused features by Eq. (9))
    $S \leftarrow \Phi_w(\widetilde{\mathcal{F}}^{\mathtt{U}})$ (generate diffusion prompts)
    $L_1([\mathrm{c}_0, S, \mathrm{E}_0]) \leftarrow$ (insert into F_δ by Eq. (10))
    $\mathcal{L}_{CE}(x^{\mathtt{U}}, S, y) \leftarrow$ (CrossEntropy loss by Eq. (11))
    // Momentum Alignment Regulating
    $v_m \leftarrow \alpha v + (1 - \alpha) v_m$ (get momentum classifier)
    $\mathcal{L}_{SIM}(x^{\mathtt{U}}, S, x) \leftarrow$ (feature consistency Eq. (12))
    $\mathcal{L}_{DIS}(x^{\mathtt{U}}, S, x) \leftarrow$ (logit consistency Eq. (13))
    $\mathcal{L} = \mathcal{L}_{CE} + \mathcal{L}_{SIM} + \mathcal{L}_{DIS}$ (overall loss Eq. (14))
  end for
end for

INFERENCE PROCESS OF COMOMDP

for each test dataset loader $\mathcal{D}_{\mathtt{test}}$ **do**
  $x \leftarrow \mathcal{D}_{\mathtt{test}}$ (get test samples)
  $x^{\mathtt{U}} \leftarrow \mathtt{U}_p + \hat{x}$ (get prompted images)
  $\mathtt{Z}_{\mathtt{test}} \leftarrow$ (title description of task dataset)
  $\mathtt{T}_{\mathtt{test}} \leftarrow$ 'a photo of a $\{\mathtt{Z}_{\mathtt{test}}\}$' (get text prompts)
  $\widetilde{\mathcal{F}}^{\mathtt{U}} \leftarrow \epsilon_\theta(\boldsymbol{\varepsilon}(x^{\mathtt{U}}), \boldsymbol{\varphi}(\mathtt{T}_{\mathtt{test}}), t = 0)$ (get fused features)
  $S_{\mathtt{test}} \leftarrow \Phi_w(\widetilde{\mathcal{F}}^{\mathtt{U}})$ (generate diffusion prompts)
  $\mathtt{y}_{\mathtt{pred}} \leftarrow F_\delta(x^{\mathtt{U}}, S_{\mathtt{test}})$ (get prediction results)
end for

C. Momentum Alignment Regulating

During training, CoMoDP maximizes the likelihood of the correct label y_i for the image x_i by Eq. (11). However, prompts tend to overfit downstream data distributions and lead to the loss of the model's generalization ability. To address this issue,

we introduce *Momentum Alignment Regulating* (MAR), a self-regulating strategy to restrict optimization regions of prompts in the feature and logit spaces. First, MAR imposes an explicit representation consistency constraint between prompt features and pre-trained features. The pre-trained model F_δ consists a backbone b_u and a classifier h_v ($\delta = \{u, v\}$). Given an image x_i, we denote $f_i = b_u(x_i)$ as embeddings of x_i, $\ell_i = h_v(f_i)$ as the logits of x_i. As usual, we use the feature $f_i^0 = b_u^0(x_i)$ of the `[cls]` token from f_i with the dimension $1 \times dim$. We condition the `[cls]` token's features of the prompted image $x_i^{\mathtt{U}}$ to be close to the features of the original image x_i:

$$\mathcal{L}_{SIM} = \sum_{d=1}^{dim} ||b_u^0(x_i^{\mathtt{U}}, S_i) - b_u^0(x_i)||_2, \tag{12}$$

where d indexes the dimension of the feature output, we utilize the L_2 loss to impose the feature level consistency. Similarly, for the logit level, we devise a momentum classifier h_{v_m} of h_v by taking the moving average of parameters expressed as $v_m \leftarrow \alpha v + (1-\alpha)v_m$, where v_m is initialized to v and α is a momentum coefficient (default set as 0.99). We regulate the prompted logits distribution by minimizing Kullback-Leibler divergence D_{KL} as:

$$\mathcal{L}_{DIS} = D_{\mathrm{KL}}(\log(\sigma(h_v(f_i^{\mathtt{U}}))), \sigma(h_{v_m}(f_i))), \tag{13}$$

where σ denotes the softmax function and $f_i^{\mathtt{U}} = b_u(x_i^{\mathtt{U}}, S_i)$, the parameters v are updated by back-propagation and v_m does not participate in updates. h_{v_m} is actually a constrained version of h_v to avoid parameters overfitting data distributions, where the pseudo-logits $h_{v_m}(f_i)$ being used as additional supervision during training. The training objective of CoMoDP becomes:

$$\mathcal{L} = \mathcal{L}_{CE} + \mathcal{L}_{SIM} + \mathcal{L}_{DIS}. \tag{14}$$

D. Summary and Advantages

CoMoDP first inserts the unified prompts $\mathtt{U}_p$ into an image x_i via the UVPD module to get the prompted image $x_i^{\mathtt{U}}$. Then, the UNet decoder of a pre-trained SD model is used to extract diffusion features, and further obtain the fused features $\widetilde{\mathcal{F}}_i^{\mathtt{U}}$ of the prompted image $x_i^{\mathtt{U}}$. Finally, the diffusion prompts S_i is generated by the CM-Net Φ_w based on $\widetilde{\mathcal{F}}_i^{\mathtt{U}}$, which is further inserted into the visual encoder F_δ via the DFPS module. *For the training process*, equipped with the MAR strategy, we use the training objective in Eq. (14) to optimize $\mathtt{U}_p$ and Φ_w. *For the inference process*, $\mathtt{T}_{y_i}$ in Eq. (8) is agnostic, so we use the description of the dataset for all test images, *e.g.*, for FGVC-Aircraft [59] dataset we set text prompt as $\mathtt{T}_{y_i}$='*a photo of a aircraft*' in Eq. (8) to evaluate our method through the learned $\mathtt{U}_p$ and Φ_w. The overview of CoMoDP is shown in Fig. 2, and the pseudo-code flow is provided in Algorithm 1.

The advantages of our method: (i) *Efficiency*: we set $\mathtt{U}_p$ and Φ_w as trainable in CoMoDP, which only accounts for 3.06% of total parameters. Besides, as shown in Fig. 1 (b), there are no excessive parameters involved compared to other visual tuning methods. Moreover, Φ_w simulates a single diffusion prompt by inference once. Our method achieves a better trade-off between performance and efficiency. (ii) *Modularity*: the modularity of CoMoDP indicates its broad range of applications. The UVPD and DFPS modules can be easily loaded to previous methods on the vision tasks. Beyond this, it can also be incorporated with text-prompt methods by applying the diffusion prompts to align the text features for prompt learning. (iii) Overhead: the diffusion feature extraction in our DFPS module involves running the UNet decoder to process images and generate intermediate feature maps. While the UNet introduces non-negligible overhead, the techniques in CoMoDP like PCA and single-pass inference also help mitigate it: 1) we reduce dimensionality via PCA, which lowers memory and computational costs; 2) the context modulator network generates diffusion prompts with a single forward pass, avoiding repeated UNet evaluations during training. We also conducted a quantitative analysis of the overhead: 1) Tunable Parameters: We set the $\mathtt{U}_p$ and Φ_w as trainable in CoMoDP, which only accounts for 3.06% of total parameters (151M). 2) Communication Overhead: The total size of unified prompts and diffusion prompts is 23.8KB, which reduces the communication volume by about 25,000 times compared with the CLIP full model (600MB). 3) Computational Overhead: Let's take the forward process as an example, the total FLOPs of the full CLIP model is 1.01×10^8 (single layer is 8.42×10^6), the total FLOPs of our CoMoDP is 1.26×10^8 (single layer is 10.46×10^6). Despite the overhead still existing, CoMoDP offers a better trade-off between performance and efficiency.

V. Experiments

A. Experimental Setups

Following [11], [12], our method is mainly evaluated in the three settings: generalization from base to new classes, cross-dataset evaluation, and domain generalization. The setup as:

Datasets: For the first two settings, we evaluate on 11 standard datasets, including ImageNet [63], Caltech101 [64], OxfordPets [65], StanfordCars [66], Flowers102 [67], Food101 [68], FGVC-Aircraft [59], SUN397 [69], UCF101 [70], DTD [71], and EuroSAT [72]. For the domain generalization setting, we use ImageNet as the source domain and use ImageNet-V [73], ImageNet-S [74], ImageNet-A [75], ImageNet-R [76] as four target domains. For few-shot experiments, we follow [11] to select samples for training and evaluate on entire test set.

Baselines: We compare our method with CoOp [9], CoCoOp [11], MaPLe [12], GalLoP [61], PromptSRC [46], CoPL [62], and visual tuning-based VPT [15] and GatedPT [60], and also large-scale zero-shot methodology CLIP [2].

Training Details: We use ViT-B/16 as our basic model for the visual encoder of CLIP, VPT, GatedPT, our CoMoDP, for other prompt-based methods we use ViT-B/16 CLIP as the model. The prompt token length of CoOp and CoCoOp is 16, CoPL is 4, MaPLe is 9, GalLoP is 8 (local and global tokens), PromptSRC is 4, and GatedPT is 120 tokens on the visual encoder. All our experiments are trained with a batch size of 64 for 200 epochs on two 32 GB Tesla V100 GPUs. Note that we use VPT-Deep version for VPT. We set size $p = 30$ in Eq. (6) and dimension $dim = 512$ in Eq. (12), momentum coefficient α set 0.99, and we use SD-1.5 version as default. Our start learning rate is 0.01 and we use cosine learning rate scheduler, and our warm-up with a learning rate is 0.0001.

TABLE II
COMPARISON PERFORMANCE OF COMODP AND BASELINES ON 11 STANDARD DATASETS, EACH TRAINING DATASET CONSISTS OF 16 SHOTS PER CLASS.

Baselines	Protocols	ImageNet	Caltech101	OxfordPets	StandfordCars	Flowers102	Food101	FGVCAircraft	SUN397	DTD	EuroSAT	UFC101	Average
CLIP [2]	Base	72.4	96.8	91.1	63.3	72.1	90.1	27.2	69.4	53.2	56.5	70.5	69.3
	Novel	68.1	94.0	97.2	74.9	77.8	91.2	36.3	75.3	59.9	64.0	77.5	74.2
	HM	70.2	95.4	94.1	68.6	74.8	90.6	31.1	72.2	56.3	60.0	73.8	71.7
CoOp [9]	Base	76.5	98.0	93.6	78.1	97.6	88.3	40.4	80.6	79.4	92.1	84.6	82.6
	Novel	67.8	89.8	95.2	60.4	59.6	82.2	22.3	65.8	41.1	54.7	56.0	63.2
	HM	71.9	93.7	94.4	68.1	74.0	85.2	28.7	72.5	54.2	68.7	67.4	71.6
CoCoOp [11]	Base	75.9	97.9	95.2	70.5	94.8	90.7	33.4	79.7	77.0	87.5	82.3	80.4
	Novel	70.4	93.8	97.7	73.6	71.7	91.3	23.7	76.8	56.0	60.0	73.4	71.7
	HM	73.1	95.8	96.4	72.0	81.7	90.9	27.7	78.2	64.8	71.2	77.6	75.8
VPT [15]	Base	74.1	98.3	94.7	74.4	95.4	91.2	42.3	77.8	79.2	92.2	85.7	82.3
	Novel	60.6	90.8	89.3	55.2	58.8	81.3	35.9	62.8	41.6	60.5	62.1	63.6
	HM	67.2	94.0	92.1	64.5	76.8	86.0	38.8	70.1	60.5	76.4	73.6	70.7
GatedPT [60]	Base	75.2	98.2	95.1	71.7	92.5	92.8	41.9	78.3	80.1	94.5	85.9	82.4
	Novel	65.6	88.9	91.9	60.0	61.3	81.4	36.9	74.5	48.1	61.0	66.7	66.9
	HM	70.1	92.7	92.9	66.3	77.5	86.9	39.0	75.7	64.2	78.9	75.8	71.5
MaPLe [12]	Base	76.6	97.7	95.4	72.9	95.9	90.7	37.4	80.8	80.3	94.0	83.0	82.2
	Novel	70.5	94.3	97.7	74.0	72.4	92.0	35.6	78.7	59.1	73.2	78.6	75.1
	HM	73.4	96.0	96.5	73.4	82.5	91.3	36.5	79.7	68.1	82.3	80.7	78.5
PromptSRC [46]	Base	77.6	98.1	95.3	78.2	98.0	90.6	42.7	82.6	83.3	92.9	87.1	84.2
	Novel	70.7	94.0	97.3	74.9	76.5	91.5	37.8	78.4	62.9	73.9	78.8	76.1
	HM	74.0	96.0	96.3	76.5	85.9	91.1	40.1	80.5	71.7	82.4	82.7	79.9
GalLoP [61]	Base	75.1	96.7	94.1	69.2	96.8	89.5	38.3	82.2	85.0	90.1	84.9	81.9
	Novel	70.8	93.7	96.7	75.5	70.9	90.2	30.2	76.5	50.6	34.3	76.0	74.6
	HM	73.2	95.5	95.2	72.3	83.0	90.9	34.3	80.7	70.6	60.4	77.9	75.7
CoPL [62]	Base	77.8	98.1	95.6	70.7	96.1	90.9	36.1	80.2	78.0	89.2	83.1	81.4
	Novel	71.3	94.9	97.8	74.4	72.1	91.4	31.3	77.3	50.0	34.2	76.6	75.6
	HM	74.5	96.5	96.7	72.6	84.1	91.5	33.7	78.7	64.0	61.7	79.8	78.5
CoMoDP	Base	**79.8**	**98.8**	**96.2**	**81.3**	97.8	92.1	**45.8**	82.0	84.5	**96.1**	**89.0**	**85.7**
	Novel	**73.9**	**96.0**	**98.2**	**80.5**	**81.6**	**92.9**	**40.5**	**80.8**	**68.0**	72.7	77.9	**78.9**
	HM	**76.7**	**97.5**	**97.2**	**81.0**	**89.2**	**92.2**	**43.1**	**81.9**	**73.3**	**84.1**	**83.6**	**81.8**

B. Base-to-Novel Knowledge Transfer

As shown in Tab. II, following the implementations of [11], we report *Base* and *Novel* class accuracy and their harmonic mean (*HM*) averaged over 3 runs on the above-mentioned 11 standard datasets. While training is only conducted on the base classes. During testing we transfer the learned knowledge to classify novel classes as well as base classes. From Tab. II, we observe that the performance of CoOp degrades if compared to large-scale zero-shot CLIP for novel classes. While conditional prompts improve the performance of CoCoOp, it still fails to generalize to novel classes with dynamic distributions (in most cases, CLIP outperforms CoCoOp, 74.2% v.s. 71.7%).

In comparison with these baselines, our proposed CoMoDP shows superior performance on all 11 datasets. Moreover, the harmonic mean values for all datasets are greater than current state-of-the-art model PromptSRC (79.9% v.s. 81.8%), which indicates that learned unified prompts and diffusion prompts are more generalizable across domains and datasets. For the relatively complex datasets such as EuroSAT [72] and UCF101 [70], where satellite localization is hard to align, for the base classes, CoMoDP exceeds CLIP by 39.6% and 18.5%.

C. Cross-Dataset Knowledge Transfer

We further demonstrate that CoMoDP has the potential to transfer beyond a single dataset, as shown in Tab. III, where we train the model on the ImageNet dataset (source) and evaluate its performance on the rest of other 10 datasets. In Tab. III, we can observe that CoMoDP achieves consistent improvements compared to the baselines, even in the more fine-grained or specialized datasets. For the complex dataset as FGVCAircraft (containing various textures), CoMoDP still exhibits stronger generalizability and transferability than other baselines. Furthermore, despite being trained only on the object classification dataset (ImageNet), CoMoDP effectively transfer knowledge to texture recognition task on DTD dataset (exceeding CoCoOp by 6.25%), showing the necessity of dynamically modulating the context for each input sample via the diffusion prompts. These results highlight that CoMoDP is able to learn the fine-grained semantics of the visual concepts in the images for task adaptation.

D. Domain Generalization Evaluation

Analogous to the cross-dataset generalization evaluation, we directly transfer the source model trained on the ImageNet (IN) to four cross-domain datasets (IN-V, IN-S, IN-A, and IN-R). As illustrated in Tab. IV, our CoMoDP outperforms state-of-the-art baselines on all of the tested datasets, which indicates the superior generalization ability of CoMoDP in addressing domain shifts, displaying its effectiveness in adapting to the diverse and challenging data scenarios. By learning the unified and diffusion prompts, our CoMoDP is able to uncover the subjective image-semantics mapping, making CoMoDP more robust while dealing with out-of-distribution datasets.

E. Ablation Studies and Analysis

To thoroughly analyze the efficacy of essential modules in our method, we perform an ablation study to investigate the UVPD and DFPS. We present the quantitative result in Tab. V, the first row refers to the zero-shot CLIP. From Tab. V, we are able to perceive three points: (i) UVPD leads to significant performance improvements against the baseline on different

TABLE III
COMPARISON OF COMODP WITH BASELINES ON CROSS-DATASET EVALUATION, SOURCE IS IMAGENET DATASET, TARGET IS OTHER 10 DATASETS.

Baselines	Source	Target										
	ImageNet	Caltech101	OxfordPets	StandfordCars	Flowers102	Food101	FGVCAircraft	SUN397	DTD	EuroSAT	UFC101	**Average**
CoOp [9]	71.51	93.70	89.14	64.51	68.71	85.30	18.47	64.15	41.92	46.39	66.55	63.88
CoCoOp [11]	71.02	**94.43**	90.14	65.32	71.88	86.06	22.94	67.36	45.73	45.37	68.21	65.74
VPT [15]	69.43	93.53	89.73	64.43	68.50	84.77	24.17	66.53	45.57	36.23	65.93	63.94
GatedPT [60]	70.59	93.68	90.26	65.10	69.05	85.86	24.40	64.51	45.80	39.31	67.71	64.56
MaPLe [12]	70.72	93.53	90.49	65.57	72.23	86.20	24.74	67.01	46.49	48.06	68.69	66.30
PromptSRC [46]	71.27	93.60	90.25	65.70	70.25	86.15	23.90	67.10	46.87	45.50	68.75	65.81
GalLoP [61]	71.31	93.89	89.78	64.45	73.12	86.69	23.38	67.50	46.43	45.28	69.20	65.97
CoPL [62]	71.78	93.95	90.36	65.81	74.62	87.36	25.80	68.19	46.77	43.35	70.30	66.65
CoMoDP	**72.85**	93.71	**91.25**	**70.39**	**75.18**	**88.25**	**26.21**	**68.75**	**51.98**	**55.52**	**74.41**	**69.56**

TABLE IV
COMPARISON OF COMODP WITH BASELINES IN THE DOMAIN GENERALIZATION SETTING ('IN' REFERS TO IMAGENET).

Baselines	Source	Target				
	IN	IN-V	IN-S	IN-A	IN-R	**Average**
CoOp [9]	71.51	64.20	47.99	49.71	75.21	59.27
CoCoOp [11]	71.02	64.07	48.75	50.63	76.18	59.91
VPT [15]	69.43	62.80	48.03	45.90	75.83	58.14
GatedPT [60]	70.59	63.49	48.96	47.20	75.67	58.83
MaPLe [12]	70.72	64.07	49.15	50.90	76.98	60.27
PromptSRC [46]	71.27	64.35	49.55	50.90	77.80	60.65
GalLoP [61]	71.31	66.50	49.48	50.37	77.83	61.04
CoPL [62]	71.78	65.20	49.42	50.81	78.63	61.02
CoMoDP	**72.85**	**67.42**	**52.75**	**55.59**	**80.63**	**64.09**

TABLE V
ABLATION STUDY OF KEY MODULES UVPD AND DFPS ('OPETS' REFERS TO OXFORDPETS, ..., 'FGVCA' REFERS TO FGVCAIRCRAFT).

UVPD	DFPS	Fine-Grained Classification Datasets					
		OPets	SCars	F102	F101	FGVCA	**Average**
✘	✘	88.43	65.58	70.70	84.80	24.85	66.87
✔	✘	90.17	75.26	82.32	85.39	39.41	74.51
✘	✔	92.85	83.28	90.98	87.03	48.60	80.54
✔	✔	94.63	89.22	97.69	87.55	58.75	**85.56**

TABLE VI
COMPARISON OF DIFFERENT SIZES AND TEMPLATES IN UNIFIED PROMPTS.

Size p	ImageNet			Caltech101		
	tem_{ran}	tem_{fix}	tem_{pad}	tem_{ran}	tem_{fix}	tem_{pad}
$p = 10$	68.96	68.25	70.66	92.54	92.62	94.33
$p = 20$	70.42	69.75	71.96	93.46	93.08	94.92
$p = 30$	71.29	70.81	**72.85**	94.47	94.28	**95.49**
$p = 40$	70.21	70.19	72.05	92.17	90.51	93.32

TABLE VII
COMPARISON OF DIFFERENT TIMESTEP t (EQ. (8)), L-TH LAYER'S FEATURES, AND FUSED FEATURES IN DIFFUSION PROMPTS.

Timestep t	ImageNet					
	$L = 1$	$L = 2$	$L = 3$	$L = 4$	Fused	**Average**
$t = 0$	68.36	70.82	71.32	70.51	**72.85**	70.77
$t = 100$	68.40	70.47	71.10	70.35	72.67	70.59
$t = 200$	68.19	70.05	70.84	70.09	72.50	70.34
$t = 500$	67.93	69.71	70.16	69.73	72.12	69.93
$t = 999$	67.70	69.43	69.89	69.65	71.83	69.70

tasks. This demonstrates that the UVPD module can produce generalizable task-specific domain knowledge in uniform pixel space; (ii) we notice solid gains by incorporating the DFPS module. This indicates the importance of extracting explicit semantic guidance from the diffusion models to enhance visual prompting; (iii) combining UVPD and DFPS modules achieves better performance. This supports our motivation to unleash joint dataset-level generalization and instance-level adaptation in visual prompting. *Next, we analyze each module in detail:*

Unified Visual Prompt Designer: The pipeline of UVPD is to shrink images and then harmonize the unified prompts, as shown in Fig. 3. We investigate the importance of preserving representations of the original image: we explore the effects of different prompt sizes p and templates in the unified prompts $\mathtt{U}_p$, as shown in Tab. VI, we explore three templates: pixel patch at random location (tem_{ran}), pixel at fixed location (tem_{fix}), and padding (tem_{pad}). From the Tab. VI, we can observe that using the padding template yields better performance across different prompt sizes p, showing the importance of keeping extra and independent unified prompts and the original image non-overlapping. Here we use $p = 30$ and tem_{pad} as default.

Diffusion Prompt Simulator: The main steps of DFPS are extracting fused diffusion features and generating diffusion prompts. We analyze the effects of timestep t and L-th layer's features in Tab. VII, where the results show that our method is not sensitive to $t \in [0, 200]$, we set $t = 0$ and fused features as default for all our experiments. We find lower L might lead to worse performance due to the low spatial resolution, extremely large L might focus too much on the texture details and lose semantics, we choose fused features $\widetilde{\mathcal{F}}$ in Eq. (9) as default.

Momentum Alignment Regulating: CoMoDP enhances the model's generalization ability in a self-regulating way: feature space ($\mathcal{L}_{SIM}$ in Eq. (12)) and logit space ($\mathcal{L}_{DIS}$ in Eq. (13)). Tab. VIII indicates the effectiveness and importance of different loss objectives in Eq. (14). Experimental results show that

TABLE VIII
ABLATION STUDY OF $\mathcal{L}_{CE}$, $\mathcal{L}_{SIM}$, AND $\mathcal{L}_{DIS}$ LOSSES IN EQ. (14).

$\mathcal{L}_{CE}$	$\mathcal{L}_{SIM}$	$\mathcal{L}_{DIS}$	ImageNet	Caltech101	**Average**
✔	✘	✘	71.37	93.56	82.46
✔	✔	✘	72.20	94.62	83.41
✔	✘	✔	71.89	94.29	83.09
✔	✔	✔	72.85	95.49	**84.17**

TABLE IX
ABLATION STUDY OF INJECTING MULTI-LAYER DIFFUSION PROMPTS ($L_1 \sim L_3$ MEANS INSERTING DIFFUSION PROMPTS AT LAYERS L_1 TO L_3), 'OPETS' REFERS TO OXFORDPETS, ..., 'FGVCA' REFERS TO FGVCAIRCRAFT.

Layers	OPets	SCars	F102	F101	FGVCA	**Average**
$L_1 \sim L_3$	94.78	89.62	97.71	87.80	58.93	85.76
$L_1 \sim L_6$	94.97	90.05	97.82	88.41	59.12	86.07
$L_1 \sim L_9$	95.13	90.70	97.91	88.69	59.47	86.38
$L_1 \sim L_{12}$	95.21	90.79	97.95	89.13	59.61	86.53

the three losses contribute positively to final accuracy: $\mathcal{L}_{CE}$ is the standard cross-entropy loss between predicted logits and ground-truth labels to ensure discriminative learning; $\mathcal{L}_{SIM}$ constrains the difference between the `[cls]` token's representations of the original image and the prompted image, which helps to preserve the semantic content; $\mathcal{L}_{DIS}$ uses the momentum classifier acts as a temporal ensemble, introducing smoothness and stability in predictions to prevent prompt-induced over-confidence. In addition, $\mathcal{L}_{SIM}$ achieves higher gain compared to $\mathcal{L}_{DIS}$, indicating that $\mathcal{L}_{SIM}$ can better guide the learning trajectory of the prompts.

Multi-layer Diffusion Prompts: We conduct experiments to explore injecting the diffusion prompts at each layer of the visual encoder based on different CM-Net, similar to VPT-Deep. As shown in Tab. IX, inserting diffusion prompts at multiple layers of the visual encoder can further improve the performance. Actually, this is predictable: 1) On the one hand, injecting diffusion prompts into deeper layers could capture hierarchical semantic features, enhancing adaptation to input-specific details. 2) On the other hand, the deeper layers in the visual encoder focus on high-level semantics to enable dynamic modulation at different abstraction levels.

Different Diffusion Weights: Since our CoMoDP is built on pre-trained diffusion models, it is necessary to investigate how the pre-trained weights would affect the performance of our CoMoDP. In our previous experiments, we have used the released Stable Diffusion Version 1.5 (SD-1.5). We now compare different releases of stable diffusion models (SD-1.4 and SD-2.0) and SD-XL [78], as shown in Fig. 5. The more powerful diffusion models like SD-XL could enhance the semantic richness of the diffusion prompts, providing more informative and robust instance-level guidance, further improving the generalization of downstream tasks. These results also show that the success of our CoMoDP is based on the learned visual-language knowledge rather than the large capacity of the diffusion model. The upward trend demonstrates the scaling ability of our method, showing that a stronger diffusion model can further improve the performance.

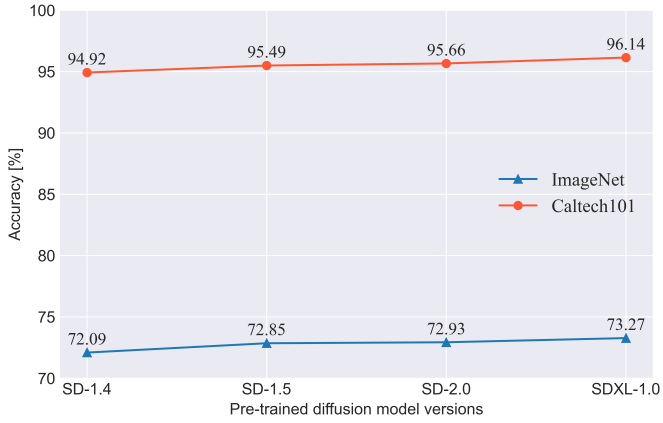

Fig. 5. The impact of different diffusion versions on the performance of CoMoDP in the ImageNet and Caltech101 datasets.

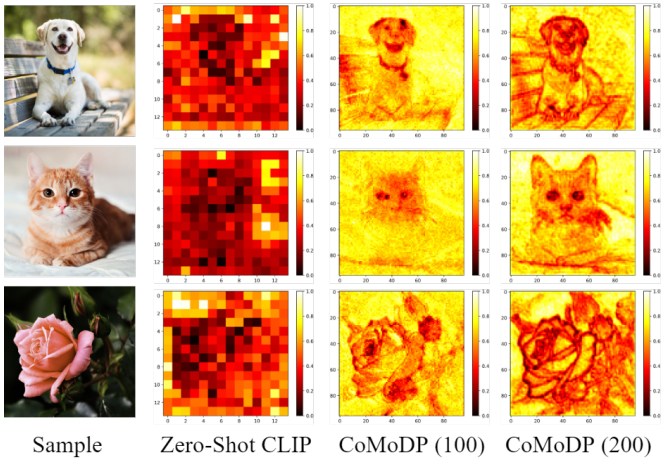

Fig. 6. Visualization on attention layer map via Grad-CAM [77] (gradient-weighted class activation mapping). From left to right, original image, zero-shot CLIP, CoMoDP with epochs 100, CoMoDP with epochs 200.

Visualization: We visualize the attention layer map based on Grad-CAM [77] (gradient-weighted class activation mapping) at the 12^{th} block of the visual encoder, as shown in Fig. 6. We can observe that CLIP's features are not discriminative and do not allow for accurate semantic perception, while our CoMoDP accurately segments the object of interest and attends to the object's boundary. This also suggests the learnable unified and diffusion prompts play a role in adjusting attention.

VI. CONCLUSION

We propose CoMoDP, a dual-visual prompting scheme with context-modulated diffusion prompts, by leveraging complementary advantages of two modules: UVPD provides unified domain knowledge at the pixel level and DFPS offers explicit semantic guidance at the embedding level. The superiority of CoMoDP has been thoroughly validated with popular counterparts in various scenarios, including generalization from base to novel, cross-dataset transfer, domain generalization. We also believe that text-guided generative models other than diffusion models can also fit in CoMoDP, which we leave to future work.

ACKNOWLEDGMENTS

This work is supported by the scholarship from the China Scholarship Council (CSC) while the first author pursues his Ph.D. degree at University of Wollongong. This work was also partially supported by the Australian Research Council (ARC) Linkage Project under Grant LP210300009 and LP230100083.

REFERENCES

[1] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *ICLR*, 2021.
[2] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh *et al.*, "Learning transferable visual models from natural language supervision," in *ICML*, 2021, pp. 8748–8763.
[3] A. Sanghi, H. Chu, J. G. Lambourne *et al.*, "Clip-forge: Towards zero-shot text-to-shape generation," in *CVPR*, 2022, pp. 18 603–18 613.
[4] Z. Zhou, Y. Lei, B. Zhang, L. Liu, and Y. Liu, "Zegclip: Towards adapting clip for zero-shot semantic segmentation," in *CVPR*, 2023, pp. 11 175–11 185.
[5] S. Martin, Y. Huang, F. Shakeri, J.-C. Pesquet *et al.*, "Transductive zero-shot and few-shot clip," in *CVPR*, 2024, pp. 28 816–28 826.
[6] K. Zhang, Y. Yang, J. Yu, H. Jiang *et al.*, "Multi-task paired masking with alignment modeling for medical vision-language pre-training," *IEEE Transactions on Multimedia*, vol. 26, pp. 4706–4721, 2023.
[7] S. Wu, X. Fu, F. Wu, and Z.-J. Zha, "Vision-and-language navigation via latent semantic alignment learning," *IEEE Transactions on Multimedia*, vol. 26, pp. 8406–8418, 2024.
[8] S. Shen, S. Yang, T. Zhang, B. Zhai *et al.*, "Multitask vision-language prompt tuning," in *WACV*, 2024, pp. 5656–5667.
[9] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Learning to prompt for vision-language models," *IJCV*, vol. 130, no. 9, pp. 2337–2348, 2022.
[10] J. Bang, S. Ahn, and J.-G. Lee, "Active prompt learning in vision language models," in *CVPR*, 2024, pp. 27 004–27 014.
[11] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Conditional prompt learning for vision-language models," in *CVPR*, 2022, pp. 16 816–16 825.
[12] M. U. Khattak, H. Rasheed, M. Maaz, S. Khan, and F. S. Khan, "Maple: Multi-modal prompt learning," in *CVPR*, 2023, pp. 19 113–19 122.
[13] P. Liu, W. Yuan, J. Fu, Z. Jiang *et al.*, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–35, 2023.
[14] L. Zhao, Q. Li, A. Tariq *et al.*, "Nlp based on prompt-based learning and prompting methods: a survey," in *AIAM*, 2023, pp. 255–259.
[15] M. Jia, L. Tang, B.-C. Chen, C. Cardie *et al.*, "Visual prompt tuning," in *ECCV*, 2022, pp. 709–727.
[16] H. Bahng, A. Jahanian, S. Sankaranarayanan, and P. Isola, "Exploring visual prompts for adapting large-scale models," *arXiv preprint arXiv:2203.17274*, 2022.
[17] L. Liu, N. Wang, D. Liu, X. Yang *et al.*, "Towards specific domain prompt learning via improved text label optimization," *IEEE Transactions on Multimedia*, vol. 26, pp. 10 805–10 815, 2024.
[18] Y. Zeng, N. Han, K. Pan, and Q. Jin, "Temporally language grounding with multi-modal multi-prompt tuning," *IEEE Transactions on Multimedia*, vol. 26, pp. 3366–3377, 2023.
[19] A. Webson and E. Pavlick, "Do prompt-based models really understand the meaning of their prompts?" in *ACL*, 2022, pp. 2300–2344.
[20] X. Wang, K. Zhou, J.-R. Wen, and W. X. Zhao, "Towards unified conversational recommender systems via knowledge-enhanced prompt learning," in *KDD*, 2022, pp. 1929–1937.
[21] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *NeurIPS*, 2020, pp. 6840–6851.
[22] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *ICML*, 2021, pp. 8162–8171.
[23] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *CVPR*, 2022, pp. 10 684–10 695.
[24] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in *ICCV*, 2023, pp. 4195–4205.
[25] A. Li, M. Prabhudesai, S. Duggal, E. Brown *et al.*, "Your diffusion model is secretly a zero-shot classifier," in *ICCV*, 2023, pp. 2206–2217.
[26] K. Clark and P. Jaini, "Text-to-image diffusion models are zero shot classifiers," in *NeurIPS*, 2024, pp. 58 921–58 937.
[27] L. Tang, M. Jia, Q. Wang, C. P. Phoo *et al.*, "Emergent correspondence from image diffusion," in *NeurIPS*, 2023, pp. 1363–1389.
[28] J. Zhang, C. Herrmann, J. Hur, L. Polania Cabrera *et al.*, "A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence," in *NeurIPS*, 2023, pp. 45 533–45 547.
[29] J. Zhang, J. Huang, S. Jin *et al.*, "Vision-language models for vision tasks: A survey," *IEEE TPAMI*, vol. 46, no. 8, pp. 5625–5644, 2024.
[30] P. Zhang, X. Li, X. Hu *et al.*, "Vinvl: Revisiting visual representations in vision-language models," in *CVPR*, 2021, pp. 5579–5588.
[31] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *ICML*, 2023, pp. 19 730–19 742.
[32] M. Maniparambil, R. Akshulakov, Y. A. D. Djilali *et al.*, "Do vision and language encoders represent the world similarly?" in *CVPR*, 2024, pp. 14 334–14 343.
[33] W. Wang, H. Bao, L. Dong, J. Bjorck *et al.*, "Image as a foreign language: Beit pretraining for vision and vision-language tasks," in *CVPR*, 2023, pp. 19 175–19 186.
[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.
[35] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *ACL*, 2019, pp. 4171–4186.
[36] J. Hu, W. Xia, X. Zhang, C. Fu *et al.*, "Enhancing sequential recommendation via llm-based semantic embedding learning," in *ACM Web Conference*, 2024, pp. 103–111.
[37] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," in *NeurIPS*, 2019, pp. 13–23.
[38] W. Kim, B. Son *et al.*, "Vilt: Vision-and-language transformer without convolution or region supervision," in *ICML*, 2021, pp. 5583–5594.
[39] J. Li, R. Selvaraju, A. Gotmare, S. Joty *et al.*, "Align before fuse: Vision and language representation learning with momentum distillation," in *NeurIPS*, 2021, pp. 9694–9705.
[40] M. Zhou, L. Zhou, S. Wang, Y. Cheng *et al.*, "Uc2: Universal cross-lingual cross-modal vision-and-language pre-training," in *CVPR*, 2021, pp. 4155–4165.
[41] Z. Gan, L. Li, C. Li, L. Wang *et al.*, "Vision-language pre-training: Basics, recent advances, and future trends," *Foundations and Trends in Computer Graphics and Vision*, vol. 14, no. 3, pp. 163–352, 2022.
[42] D. Li, J. Li, H. Li, J. C. Niebles, and S. C. Hoi, "Align and prompt: Video-and-language pre-training with entity prompts," in *CVPR*, 2023, pp. 4953–4963.
[43] J. Wu, T. Yu, R. Wang, Z. Song *et al.*, "Infoprompt: Information-theoretic soft prompt tuning for natural language understanding," in *NeurIPS*, 2024, pp. 61 060–61 084.
[44] Y. Hirohashi, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, "Prompt learning with one-shot setting based feature space analysis in vision-and-language models," in *CVPR*, 2024, pp. 7761–7770.
[45] Z. Wang, Z. Zhang, C.-Y. Lee, H. Zhang *et al.*, "Learning to prompt for continual learning," in *CVPR*, 2022, pp. 139–149.
[46] M. U. Khattak, S. T. Wasim, M. Naseer, S. Khan *et al.*, "Self-regulating prompts: Foundational model adaptation without forgetting," in *ICCV*, 2023, pp. 15 190–15 200.
[47] Z. Li, X. Li, X. Fu *et al.*, "Promptkd: Unsupervised prompt distillation for vision-language models," in *CVPR*, 2024, pp. 26 617–26 626.
[48] N. Tumanyan, M. Geyer, S. Bagon, and T. Dekel, "Plug-and-play diffusion features for text-driven image-to-image translation," in *CVPR*, 2023, pp. 1921–1930.
[49] D. A. Hudson, D. Zoran, M. Malinowski, A. K. Lampinen *et al.*, "Soda: Bottleneck diffusion models for representation learning," in *CVPR*, 2024, pp. 23 115–23 127.
[50] G. Luo, L. Dunlap, D. H. Park, A. Holynski, and T. Darrell, "Diffusion hyperfeatures: Searching through time and space for semantic correspondence," in *NeurIPS*, 2023, pp. 47 500–47 510.
[51] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *IEEE TPAMI*, vol. 45, no. 9, pp. 10 850–10 869, 2023.
[52] G. Somepalli, V. Singla, M. Goldblum, J. Geiping, and T. Goldstein, "Understanding and mitigating copying in diffusion models," in *NeurIPS*, 2023, pp. 47 783–47 803.
[53] H.-H. Wu, P. Seetharaman, K. Kumar, and J. P. Bello, "Wav2clip: Learning robust audio representations from clip," in *ICASSP*, 2022, pp. 4563–4567.
[54] A. Fang, G. Ilharco, M. Wortsman, Y. Wan *et al.*, "Data determines distributional robustness in contrastive language image pre-training," in *ICML*, 2022, pp. 6216–6234.
[55] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*, 2015, pp. 234–241.

[56] B. Gao, H. Gouk, Y. Yang, and T. Hospedales, "Loss function learning for domain generalization by implicit gradient," in *ICML*, 2022, pp. 7002–7016.
[57] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *ICML*, 2018, pp. 794–803.
[58] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of Operations Research*, vol. 134, pp. 19–67, 2005.
[59] S. Maji, E. Rahtu, J. Kannala, M. Blaschko *et al.*, "Fine-grained visual classification of aircraft," *arXiv preprint arXiv:1306.5151*, 2013.
[60] S. Yoo, E. Kim, D. Jung, J. Lee, and S. Yoon, "Improving visual prompt tuning for self-supervised vision transformers," in *ICML*, 2023, pp. 40 075–40 092.
[61] M. Lafon, E. Ramzi, C. Rambour, N. Audebert, and N. Thome, "Gallop: Learning global and local prompts for vision-language models," in *ECCV*, 2024, pp. 264–282.
[62] K. Goswami, S. Karanam, P. Udhayanan, K. Joseph, and B. V. Srinivasan, "Copl: Contextual prompt learning for vision-language understanding," in *AAAI*, 2024, pp. 18 090–18 098.
[63] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009, pp. 248–255.
[64] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *CVPR-W*, 2004, pp. 178–178.
[65] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. Jawahar, "Cats and dogs," in *CVPR*, 2012, pp. 3498–3505.
[66] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3d object representations for fine-grained categorization," in *ICCV*, 2013, pp. 554–561.
[67] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *ICVGIP*, 2008, pp. 722–729.
[68] L. Bossard, M. Guillaumin *et al.*, "Food-101–mining discriminative components with random forests," in *ECCV*, 2014, pp. 446–461.
[69] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," in *CVPR*, 2010, pp. 3485–3492.
[70] K. Soomro, A. R. Zamir *et al.*, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.
[71] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing textures in the wild," in *CVPR*, 2014, pp. 3606–3613.
[72] P. Helber, B. Bischke, A. Dengel, and D. Borth, "Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 12, no. 7, pp. 2217–2226, 2019.
[73] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar, "Do imagenet classifiers generalize to imagenet?" in *ICML*, 2019, pp. 5389–5400.
[74] H. Wang, S. Ge, Z. Lipton, and E. P. Xing, "Learning robust global representations by penalizing local predictive power," in *NeurIPS*, 2019.
[75] D. Hendrycks, K. Zhao, S. Basart, J. Steinhardt, and D. Song, "Natural adversarial examples," in *CVPR*, 2021, pp. 15 262–15 271.
[76] D. Hendrycks, S. Basart, N. Mu, S. Kadavath *et al.*, "The many faces of robustness: A critical analysis of out-of-distribution generalization," in *ICCV*, 2021, pp. 8340–8349.
[77] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *ICCV*, 2017, pp. 618–626.
[78] D. Podell, Z. English, K. Lacey, A. Blattmann *et al.*, "Sdxl: Improving latent diffusion models for high-resolution image synthesis," in *ICLR*, 2024. [Online]. Available: https://openreview.net/forum?id=di52zR8xgf

Huan Wang (Student Member, IEEE) received the BEng degree in Computer Science and Technology from Anhui University of Science and Technology, Anhui, China (2020). In 2023, he received the Master degree in computer science and technology from Xidian University, Xi'an, China. He is currently pursuing a Ph.D. degree with the School of Computing and Information Technology, University of Wollongong, Wollongong, Australia. His research interests include federated learning, diffusion models.

Haoran Li (Student Member, IEEE) is currently pursuing his PhD degree at the School of Computing and Information Technology, University of Wollongong. He regularly serves as a reviewer of CVPR, ICCV, ECCV, NeurIPS, AAAI, T-ITS, T-CVST, PR, etc. His research interests include computational biology and computer vision. Before joining UOW, he obtained his Master degree from University of Leeds.

Huaming Chen (Member, IEEE) received the Ph.D. degree from the University of Wollongong, Wollongong, Australia. Currently, he is a Senior Lecturer with the School of Electrical and Computer Engineering, University of Sydney, Sydney, Australia. His research interests include software engineering/security, trustworthy AI, and applied machine learning. He regularly serves on the Program Committees of ACM MM, CCS, ACSAC, SANER, IJCAI, KDD, The Web Conference, SIAM ICDM, ECML/PKDD, and so on.

Jun Yan is currently a Professor in the School of Computing and Information Technology, Faculty of Engineering and Information Sciences, UOW, Australia. He earned his PhD from the Swinburne University of Technology, Australia, in 2005 and his MSc and BSc from Southeast University, China, in 2001 and 1998, respectively. His research interests include service computing, cloud computing, workflow management, and eBusiness.

Jiahua Shi received the Ph.D. degree in Chemical and Biomedical Engineering from Nanyang Technological University in Singapore. She is currently a FaBA Senior Research Fellow and Future Research Leader at the University of Queensland in Australia. Her research interests include biomedical imaging and neurological research.

Jun Shen (Senior Member, IEEE) received the Ph.D. degree from Southeast University, China, in 2001. He held positions with the Swinburne University of Technology, Melbourne and University of South Australia, Adelaide, before 2006. He is currently Full Professor with the School of Computing and Information Technology, University of Wollongong, Wollongong, NSW, Australia. He is a Senior Member of professional institutions, such as ACM and IEEE. He has published more than 420 papers in journals and conferences in CS/IT areas. His research interests include services and cloud computing, computational intelligence, bioinformatics, and so on. He has been an editor, the PC chair, the guest editor, for numerous journals and conferences. He was also a member of ACM/AIS Task Force on Curriculum MSIS 2016. His publications appeared at IEEE TRANSACTIONS ON SERVICE COMPUTING, IEEE TRANSACTIONS ON Parallel and distributed SYSTEMS, and many others.