

# CAPT: Cross-modal Anchored Prompt Tuning for Universal Cross-Domain Retrieval

Chanmin Park\*      Minyoung Kang\*  
 POSCO DX

{prov169, mykang}@poscodx.com

\* These authors contributed equally.

## Abstract

*Universal Cross-Domain Retrieval (UCDR) aims to retrieve relevant images from unseen domains and categories, requiring generalization under both domain and semantic shifts. Recent methods improve UCDR by learning or simulating source-domain prompt components, but such source-based adaptation may still entangle class semantics with domain-specific visual cues. We instead propose **CAPT** (Cross-modal Anchored Prompt Tuning), a parameter-efficient framework that adapts a fully frozen CLIP backbone without domain-specific prompts or backbone fine-tuning. The key module, **DSIP** (Deep Self-Instance Prompting), reads out layer-wise instance-specific visual prompts from CLIP’s frozen text-class anchors, constraining per-image adaptation to convex combinations of those anchors rather than to free visual features or source prompts. For cross-domain feature alignment, **PACE** (Per-sample Alignment via Cross-domain Gap Estimation) routes each sample between contrastive alignment and prototype anchoring using a residual domain gap, measured by the similarity difference between same-domain and cross-domain class prototypes. Routing on this gap removes the hand-tuned loss weight between the two objectives and chiefly protects UdCDR retrieval. Across DomainNet, Sketchy, and TU-Berlin, CAPT sets a new state of the art with only  $\leq 1.35M$  trainable parameters. Code will be made publicly available.*

## 1. Introduction

Cross-domain retrieval (CDR) is widely used in settings such as image search [7, 30], product recommendation [5, 9], and sketch-based retrieval [6]. Most existing methods, however, operate under a *closed-set* assumption, where the domains and classes are shared between training and testing [1, 2]. This assumption

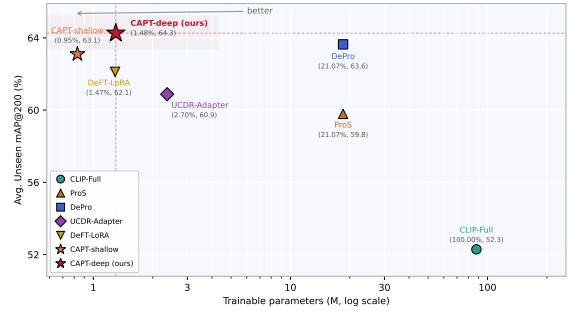


Figure 1. **CAPT is the most accurate at a comparable parameter budget.** Unseen-Gallery mAP@200 averaged over the five held-out DomainNet domains (Tab. 1, Average block) vs. trainable parameters. CAPT tops the field while training only  $\sim 1.35M$  parameters, on par with the most parameter-efficient baselines (e.g., DeFT-LoRA, 1.29M).

rarely holds in practice, as deployed retrieval systems often encounter new visual domains and new classes after deployment. Retrieval methods therefore need to remain accurate in such open-set scenarios. Universal Cross-Domain Retrieval (UCDR) [25] addresses this generalized setting, requiring accurate retrieval even when a query comes from an unseen domain, an unseen category, or both [24].

UCDR is challenging because both the visual domain and the category can shift at test time. Vision-language models such as CLIP [26] provide strong zero-shot knowledge for unseen categories, and recent methods adapt such models to UCDR through prompt tuning [13, 16, 41]. These methods improve UCDR performance, but most still learn or simulate prompt components from source domains. For example, DePro [3] decouples universal domain prompts and class prompts, while ProS [8] and UCDR-Adapter [14] construct unseen-domain or unseen-category prompts by masking or attending to seen prompts. DeFT-LoRA [36] combines a domain-invariant base expert with domain-specific experts through a gate. Although

effective, such source-based adaptation can entangle class semantics with domain-specific visual cues. This can weaken the transfer of CLIP’s zero-shot knowledge when the test domain is far from the source domains.

We take the opposite route: adaptation for UCDR without any domain prompt. Instead of encoding each source domain’s appearance into prompts, we use CLIP’s frozen text-class anchors as the source for image-specific prompt composition. Each image reads out its own visual prompt from CLIP’s frozen class-name text anchors. Therefore, prompt generation does not rely on domain-specific prompt tables, masked source prompts, or predefined style lists. We also keep the entire CLIP backbone frozen, reducing reliance on source-domain cues while preserving CLIP’s pretrained semantic structure.

We instantiate this idea as **CAPT** (Cross-modal Anchored Prompt Tuning), which has two core modules. First, **DSIP** (Deep Self-Instance Prompting) reads out instance-specific visual prompt tokens from image-conditioned frozen text-class anchors and injects them across CLIP layers. Rather than being generated freely from visual features, the prompts are composed from these frozen anchors. In this way, DSIP enables instance-specific adaptation while constraining prompt composition to CLIP’s semantic anchor set. This constraint acts on the *mixing* of the anchors and, as we verify empirically, it survives the read-out’s attention and projection. Most of DSIP’s gain comes from this structured, image-conditioned read-out; the anchors’ semantics add only a modest prior (Sec. 4.5). Second, **PACE** (Per-sample Alignment via Cross-domain Gap Estimation) aligns retrieval features across domains. PACE does not learn domain information into prompts or expert modules. Instead, it measures each sample’s residual domain gap, defined as the similarity difference between its same-domain prototype and cross-domain class prototype. This gap indicates how strongly the sample is still tied to its source domain. PACE uses this gap to route each sample between cross-domain contrastive alignment and prototype anchoring, so alignment is guided by a UCDR-specific domain-gap signal in addition to generic confidence.

With only  $\sim 1.35\text{M}$  trainable parameters, comparable to recent parameter-efficient methods, CAPT shows on five held-out DomainNet domains, Sketchy, and TU-Berlin that anchor-constrained prompt generation and measured domain-gap routing adapt a frozen CLIP beyond the training distribution. Our contributions are:

- We propose **CAPT**, a parameter-efficient UCDR framework built around **DSIP**, which adapts a fully frozen CLIP using layer-wise, instance-specific visual prompts read out from frozen text-class

anchors, without domain-specific prompts, seen-domain simulation, or backbone fine-tuning.

- We introduce **PACE**, a per-sample alignment strategy that routes each sample between cross-domain contrastive alignment and prototype anchoring using its measured *residual domain gap*, requiring no hand-tuned loss weight between the two objectives; its gain concentrates on  $\text{U}^{\text{d}}\text{CDR}$ .
- We isolate the sources of the gain with a fully frozen-backbone comparison, same-environment cross-dataset reimplementations, and routing ablations (Sec. 4).
- On DomainNet, Sketchy, and TU-Berlin, CAPT sets a new state of the art for UCDR with only  $\leq 1.35\text{M}$  trainable parameters, winning all 20 Unseen and Mixed cells against DePro.

## 2. Related Work

CAPT relates to two complementary lines of research. First, it builds on prompt learning for vision–language models, but differs by generating instance-specific visual prompts from frozen text-class anchors rather than learning source-domain prompts. Second, it connects to contrastive and prototype-based objectives for cross-domain representation learning, where our PACE module adaptively balances these two objectives using a measured residual domain gap.

### 2.1. Prompt Learning for Vision–Language Models

Prompt tuning adapts frozen visionlanguage models by learning a small set of trainable tokens instead of fine-tuning the backbone parameters. These tokens may be textual [17, 37, 40–42], visual [13, 19], or multi-modal [16], while adapter-based variants modify intermediate features rather than the input [10, 38] themselves. Beyond learning a fixed prompt table, several methods generate prompts conditioned on individual images. PGN [22] generates prompts solely from visual features; CAVPT [35] derives prompts from learned textual prompts at the input layer; HiCroPL [39] refines learned prompts hierarchically; and TPPT [23] employs frozen text prototypes as supervision targets rather than as sources for prompt generation.

CAPT differs from these approaches in a fundamental way. It generates prompts from frozen text-class anchors through an anchor-guided read-out, keeping prompt composition within the convex hull of CLIP’s text-class anchors and avoiding arbitrary prompt drift. Consequently, our contribution is not instance-conditioned prompting [40] itself, nor cross-modal prompt generation [16, 35], but anchor-guided prompt synthesis for universal cross-domain retrieval. Unlike test-time prompt tuning methods [32], which

optimize prompts separately for each query instance during inference, CAPT generates prompts in a single forward pass without any test-time optimization.

For UCDR, SnMpNet [25] first defined the problem setting. Recent CLIP-based methods adapt UCDR through domain-aware prompt- or expert-based mechanisms: DePro [3] disentangles domain and class prompts, ProS [8] and UCDR-Adapter [14] simulate unseen domains by masking seen-domain prompts, and DeFT-LoRA [36] gates domain-specific experts. Despite their differences, these methods rely on domain structures observed during training, which may limit generalization to previously unseen domains. In contrast, CAPT generates each prompt from frozen text-class anchors and therefore requires no domain-specific prompts, prompt masking, or predefined style lists; domain labels enter only PACE’s training objective (Sec. 3.4). Rather than explicitly modeling domain identities, CAPT performs anchor-constrained adaptation within the pretrained semantic space. Finally, CLIP-SBIR [29] can be viewed as a special case of domain transfer (e.g., sketch-to-photo), whereas our evaluation considers broader cross-dataset domain shifts.

## 2.2. Contrastive Learning and Class Prototypes

Contrastive learning [11, 18, 27] and prototype-based regularization are two widely adopted strategies for cross-domain representation learning. PACE combines these two objectives in a complementary manner. Its CDCM module extends memory-bank contrastive learning to the UCDR setting by treating samples of the same class but from different domains as positives, while samples of different classes and different domains serve as hard negatives. Consequently, all contrastive pairs explicitly cross domain boundaries. In parallel, CDPA performs prototype attraction [12, 20] toward flat class prototypes, the  $\ell_2$ -normalized mean of each class’s features pooled over all training domains. Several prior works have incorporated both contrastive and prototype-based objectives, including U<sup>2</sup>CDR [34] in *unsupervised* cross-domain settings. However, their combination is typically controlled by fixed hyperparameters or learnable balancing weights [4, 15]. PACE instead determines the balance on a *per-sample* basis using each query’s *residual domain gap*, the similarity difference between its same-domain and cross-domain class prototypes. Because this signal directly measures the domain shift UCDR targets, the contrastive-prototype trade-off is derived from the data rather than imposed by tuned or learned loss weights. We validate this choice directly: in Sec. 4.5 we replace the routing rule with learned alternatives, including an MLP-predicted weight, uncertainty weighting [15], and

GroupDRO [28], and find that none of them matches the measured per-sample rule.

## 3. Method

### 3.1. Problem Formulation

Universal Cross-Domain Retrieval (UCDR) aims to retrieve semantically relevant gallery images for a query whose domain or category may be unseen during training. Let  $\mathcal{D}_s$  and  $\mathcal{C}_s$  denote the seen domains and seen categories. The training set is  $\mathcal{T} = \{(x_i, y_i, d_i)\}_{i=1}^n$ , where  $x_i$  is an image,  $y_i \in \mathcal{C}_s$  is its class label, and  $d_i \in \mathcal{D}_s$  is its domain label. At test time, a query may come from an unseen domain, an unseen category, or both. We use a shared image encoder  $\Phi$  for both query and gallery images. Given an image  $x$ ,  $\Phi$  outputs an  $\ell_2$ -normalized feature  $z = \Phi(x) \in \mathbb{R}^{d'}$ . Retrieval ranks gallery features by the inner product  $z_q^\top z_g$  between query and gallery features, which equals cosine similarity. The goal is to learn  $\Phi$  from seen domains and categories while preserving generalization to unseen domains and categories.

### 3.2. Overview

We instantiate  $\Phi$  with a frozen CLIP ViT-B/32 image encoder and propose **CAPT**, which consists of two core modules. DSIP reads out layer-wise, instance-specific visual prompts from image-conditioned frozen text-class anchors, while PACE aligns the prompted features across domains, routing each sample between CDCM and CDPA by its residual domain gap.

Fig. 2 shows the overall pipeline. DSIP injects the generated prompts into each layer of the frozen encoder, and the output image feature is trained with image-text contrastive supervision and the PACE objective. During training, both CLIP encoders are frozen. We update only DSIP, a lightweight output-refinement layer, and a small text context for image-text supervision, which together amount to  $\sim 1.35\text{M}$  trainable parameters. The refinement layer is a residual MLP,  $f \leftarrow \text{norm}(f + \text{MLP}(f))$  ( $\sim 0.26\text{M}$ ), whose output is both the retrieval embedding and the feature stored in PACE’s bank; it is a standard adapter, not a contribution (isolated in Sec. 4.5). At inference time, retrieval uses only the final image embeddings. The frozen text-class anchors are used only for prompt generation, while the text targets are used only for image-text supervision.

### 3.3. Deep Self-Instance Prompting

Instead of learning prompts tied to source domains, CAPT uses **DSIP** to compose an image-specific visual prompt from frozen text-class anchors. These anchors

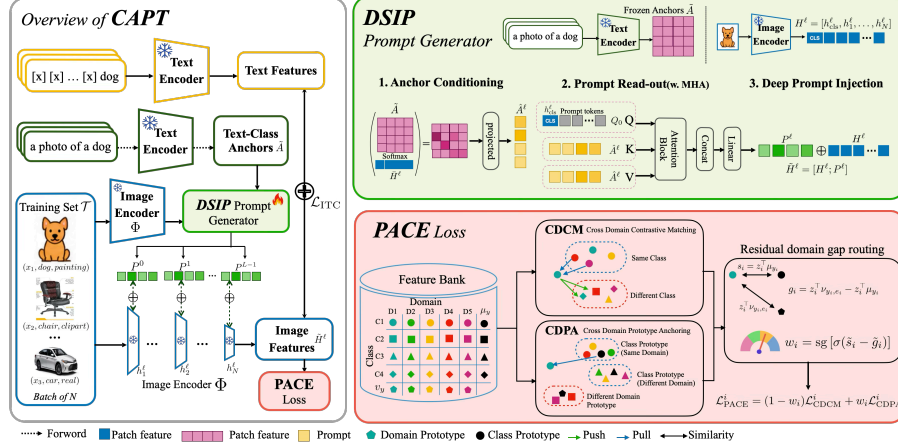


Figure 2. **Overall overview of CAPT.** CAPT adapts a frozen CLIP ViT-B/32 image encoder with two modules. DSIP reads layer-wise visual prompts from image-conditioned frozen text-class anchors and injects them into the frozen visual encoder. PACE aligns the output features by routing each sample between CDCM and CDPA by its residual domain gap.

are produced by CLIP’s frozen text encoder and provide class-level semantic guidance without domain labels or predefined style names. Thus, the prompt generator is encouraged to follow CLIP’s transferable semantic structure rather than the appearance cues of seen domains. Let  $H^\ell = [h_{\text{cls}}^\ell, h_1^\ell, \dots, h_N^\ell] \in \mathbb{R}^{(N+1) \times d}$  denote the visual token sequence at layer  $\ell$ , where  $d=768$  and  $N$  is the number of image patches. We construct the anchor bank  $A \in \mathbb{R}^{C \times d_t}$  by encoding “a photo of a [class]” for the  $C$  seen classes with the frozen CLIP text encoder, where  $d_t=512$ . Since  $A$  uses only seen class names, it leaks no unseen-class labels.

**Anchor-conditioned read-out.** DSIP first maps the frozen text anchors and the current patch tokens into a shared bottleneck space. At layer  $\ell$ , we project them as  $\tilde{A} = AW_a \in \mathbb{R}^{C \times r}$  and  $\tilde{H}^\ell = [h_1^\ell, \dots, h_N^\ell]W_h \in \mathbb{R}^{N \times r}$ . The anchors are then conditioned on the image tokens by a gated attention:

$$\hat{A}^\ell = \tilde{A} + \sigma(\gamma) \text{softmax} \left( \frac{\tilde{A}(\tilde{H}^\ell)^\top}{\sqrt{r}} \right) \tilde{H}^\ell, \quad (1)$$

where  $\sigma(\gamma)$  is a learnable scalar gate.

Static learnable queries  $Q_0 \in \mathbb{R}^{P_q \times r}$  then read out the visual prompt from the conditioned anchors:

$$P^\ell = \text{MHA}_\ell(Q_0, \hat{A}^\ell, \hat{A}^\ell) W_u^\ell \in \mathbb{R}^{P_q \times d}. \quad (2)$$

Although  $Q_0$  is shared across images,  $P^\ell$  is image-specific because  $\hat{A}^\ell$  is conditioned on the current image. Since the prompt is read from anchor-derived features, DSIP keeps adaptation close to CLIP’s semantic anchor space and avoids storing seen-domain cues in prompt parameters.

**Deep injection.** The generated prompt is appended to the token sequence at every transformer layer:

$$\tilde{H}^\ell = [H^\ell; P^\ell], \quad H^{\ell+1} = \text{CLIPBlock}^\ell(\tilde{H}^\ell), \quad (3)$$

for  $\ell = 0, \dots, L-1$ . This provides layer-wise instance adaptation while keeping the CLIP image encoder frozen. Since no domain prompt or domain label is used, the same generator is directly applied to unseen domains.

### 3.4. Per-sample Alignment via Cross-domain Gap Estimation

DSIP adapts the frozen image encoder to each input, but retrieval also requires cross-domain feature alignment. PACE maintains a global FIFO feature bank  $\mathcal{B} = \{(z_a, y_a, e_a)\}_{a=1}^M$ , where  $z_a$ ,  $y_a$ , and  $e_a$  denote the stored feature, class label, and domain label. The bank provides cross-domain positives and class prototypes beyond the current mini-batch. Unlike DSIP, PACE uses the domain label  $e_i$ , but only at training time (positives and prototypes); inference is label-free.

**Cross-domain contrastive matching.** For sample  $i$ , CDCM treats same-class samples from different domains as positives:

$$\mathcal{P}(i) = \{a : y_a = y_i, e_a \neq e_i\},$$

$$\mathcal{L}_{\text{CDCM}}^i = -\frac{1}{|\mathcal{P}(i)|} \sum_{p \in \mathcal{P}(i)} \log \frac{\exp(z_i^\top z_p / \tau_c)}{\sum_{a \in \mathcal{P}(i) \cup \mathcal{N}_K(i)} \exp(z_i^\top z_a / \tau_c)}. \quad (4)$$

If  $\mathcal{P}(i)$  is empty, this term is skipped. The denominator runs over the positives and the  $K=256$  hardest negatives  $\mathcal{N}_K(i)$  retrieved from the bank, not the full bank; this matches the implementation and the value reported in the supplement.

**Class-domain prototype anchoring.** To preserve class compactness, CDPA pulls each sample toward its class prototype. We compute  $\mu_c$  as the *flat*  $\ell_2$ -normalized mean of class- $c$  features in the bank (not

a domain-balanced average):

$$\mathcal{L}_{\text{CDPA}}^i = 1 - z_i^\top \mu_{y_i}. \quad (5)$$

CDCM promotes cross-domain spreading, while CDPA keeps each class compact around its prototype. A fixed balance is therefore not optimal for all samples.

**Residual domain-gap routing.** PACE uses two sample-wise signals. The first is prototype confidence,  $s_i = z_i^\top \mu_{y_i}$ . The second is the residual domain gap:

$$g_i = z_i^\top \nu_{y_i, e_i} - z_i^\top \mu_{y_i}, \quad (6)$$

where  $\nu_{y_i, e_i}$  is the same-domain prototype of class  $y_i$  in domain  $e_i$ . A large  $g_i$  means the sample is still closer to its own-domain prototype than to the cross-domain class prototype. We standardize both signals in the mini-batch and compute the routing weight

$$w_i = \text{sg}[\sigma(\tilde{s}_i - \tilde{g}_i)], \quad (7)$$

where  $\tilde{s}_i = (s_i - \bar{s})/\sigma_s$ ,  $\tilde{g}_i = (g_i - \bar{g})/\sigma_g$ , and  $\text{sg}[\cdot]$  denotes stop-gradient. If the same-domain prototype is unavailable,  $w_i$  falls back to the confidence-only form  $\text{sg}[\sigma((s_i - \bar{s})/\tau_w)]$ . The final PACE loss is

$$\mathcal{L}_{\text{PACE}}^i = (1 - w_i)\mathcal{L}_{\text{CDCM}}^i + w_i\mathcal{L}_{\text{CDPA}}^i. \quad (8)$$

A larger residual domain gap increases the CDCM contribution, while a confident and already aligned sample receives more CDPA. Thus, PACE uses a UCDR-specific domain-gap signal in addition to a generic confidence score, without manually tuning the balance between CDCM and CDPA.

### 3.5. Overall Objective

The overall objective combines image-text contrastive supervision with the PACE alignment loss. The image-text contrastive loss aligns the image embedding with its class text target:

$$\mathcal{L}_{\text{ITC}} = -\frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} \log \frac{\exp(z_i^\top u_{y_i}/\tau_t)}{\sum_{c=1}^C \exp(z_i^\top u_c/\tau_t)}. \quad (9)$$

where  $\mathcal{S}$  is the mini-batch and  $u_c$  is the text target for class  $c$ . The final training objective is

$$\mathcal{L} = \mathcal{L}_{\text{ITC}} + \lambda \frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} \mathcal{L}_{\text{PACE}}^i. \quad (10)$$

PACE determines the CDCM-CDPA balance for each sample, while  $\lambda$  controls the overall strength of the alignment loss. During training, both CLIP encoders are frozen. We update only DSIP, the lightweight output-refinement layer, and the text context used for the ITC targets. At inference time, retrieval uses cosine similarity between final image embeddings.

## 4. Experiments

### 4.1. Setup

We evaluate CAPT on DomainNet with a frozen CLIP ViT-B/32 image encoder. Following the standard Sn-

MpNet protocol [24, 25], we use the official class split of 245 training, 55 validation, and 45 test classes under a leave-one-domain-out setting. The Real domain is never held out and always serves as the gallery domain [25]. We report results on five held-out domains, namely clipart, sketch, infograph, painting, and quickdraw. For each held-out domain, we evaluate Unseen Gallery, Mixed Gallery, and UdCDR. Unseen Gallery and Mixed Gallery use the same unseen-domain and unseen-class queries. They differ only in the gallery. Unseen Gallery contains only unseen-class images, while Mixed Gallery additionally includes seen-class distractors. UdCDR uses unseen-domain queries from seen classes, isolating the domain shift. For all settings, we report mAP@200 and Prec@200.

### 4.2. Implementation Details

We freeze both CLIP ViT-B/32 encoders and train only DSIP, a small shared text context, and a lightweight output-refinement layer, with  $\sim 1.35\text{M}$  trainable parameters. CAPT uses  $P_q=8$  prompt tokens, bottleneck  $r=64$  and 4 heads, injected into all  $L=12$  image-encoder layers; the CoOp-style class context has length 4 and, with a per-class shift, is the only learnable text-side parameter; no domain-specific prompts are learned. The output-refinement layer adds  $\sim 0.26\text{M}$  parameters and is ablated later. We train with  $\mathcal{L} = \mathcal{L}_{\text{ITC}} + \lambda\mathcal{L}_{\text{PACE}}$ ,  $\lambda=0.5$ , router temperature  $\tau=0.1$ , a shared CDCM/CDPA bank of  $M=10,000$  features at contrastive temperature 0.07, RandAugment (2,9) at  $224 \times 224$ , Adam at a constant  $10^{-4}$  and batch size 60; all DomainNet results use the epoch 4 checkpoint and retrieval is cosine similarity between final image embeddings.

### 4.3. Main Results

Table 1 reports all five held-out domains and their average. We compare CAPT against the prompt-based ProS [8], UCDR-Adapter [14], and DeFT-LoRA [36], and the strong DePro [3] baseline, in two variants: CAPT-shallow (input-layer prompt) and CAPT-deep (all layers). Against DePro, the strongest prior method, CAPT-deep wins 21 of the 30 cells, including *every* Unseen and Mixed cell (20/20). This full sweep includes the refinement layer, and the core DSIP+PACE model still wins most Unseen and Mixed cells without it (disclosed below). On *average*, CAPT-deep leads all four Unseen and Mixed metrics. The shallow variant trails DePro on average: all-layer injection is what lifts CAPT. DePro keeps most UdCDR cells, though we win UdCDR mAP on clipart, and even on the hardest domain, quickdraw, CAPT tops both generalization protocols. The average margins are modest (Unseen

Table 1. Comparison on DomainNet (mAP@200 / Prec@200). Gray rows are earlier baselines. **Bold** and underline mark the best and runner-up among CLIP-era methods. † denotes reproduced methods. CAPT-deep entries carry  $\pm$  one standard deviation over three training seeds; baselines are single runs under their published protocols. *Significance* below.

| Query Domain   | Method                         | Venue    | UCDR                      |                           |                           |                           | U <sup>d</sup> CDR        |                           |
|----------------|--------------------------------|----------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|
|                |                                |          | Unseen Gallery            |                           | Mixed Gallery             |                           | mAP@200                   | Prec@200                  |
|                |                                |          | mAP@200                   | Prec@200                  | mAP@200                   | Prec@200                  |                           |                           |
| Clipart        | ProS <sup>†</sup> [8]          | CVPR'24  | 0.7602                    | 0.7133                    | 0.7178                    | 0.6762                    | 0.8261                    | 0.5324                    |
|                | UCDR-Adapter <sup>†</sup> [14] | WACV'25  | 0.7719                    | 0.7268                    | 0.7375                    | 0.6970                    | 0.8347                    | 0.5413                    |
|                | DeFT-LoRA [36]                 | AAAI'26  | 0.7739                    | 0.7276                    | 0.7259                    | 0.6851                    | 0.8358                    | 0.5389                    |
|                | DePro <sup>†</sup> [3]         | SIGIR'25 | 0.7906                    | 0.7488                    | 0.7349                    | 0.6992                    | 0.8555                    | <b>0.5529</b>             |
|                | CAPT-shallow                   | –        | 0.7850                    | 0.7403                    | 0.7324                    | 0.6933                    | 0.8375                    | 0.5327                    |
|                | <b>CAPT-deep (ours)</b>        | –        | <b>0.7990</b> $\pm$ .0017 | <b>0.7589</b> $\pm$ .0015 | <b>0.7436</b> $\pm$ .0012 | <b>0.7099</b> $\pm$ .0014 | <b>0.8590</b> $\pm$ .0027 | <u>0.5491</u> $\pm$ .0011 |
| Sketch         | ProS <sup>†</sup> [8]          | CVPR'24  | 0.6451                    | 0.5998                    | 0.5799                    | 0.5420                    | 0.7364                    | 0.4921                    |
|                | UCDR-Adapter <sup>†</sup> [14] | WACV'25  | 0.6570                    | 0.6084                    | 0.6044                    | 0.5652                    | 0.7422                    | 0.4957                    |
|                | DeFT-LoRA [36]                 | AAAI'26  | 0.6663                    | 0.6207                    | 0.6014                    | 0.5643                    | 0.7496                    | 0.5021                    |
|                | DePro <sup>†</sup> [3]         | SIGIR'25 | 0.6843                    | 0.6434                    | 0.6048                    | 0.5734                    | <b>0.7934</b>             | <b>0.5251</b>             |
|                | CAPT-shallow                   | –        | 0.6789                    | 0.6355                    | 0.6023                    | 0.5694                    | 0.7602                    | 0.4996                    |
|                | <b>CAPT-deep (ours)</b>        | –        | <b>0.6944</b> $\pm$ .0020 | <b>0.6527</b> $\pm$ .0021 | <b>0.6141</b> $\pm$ .0024 | <b>0.5841</b> $\pm$ .0023 | <u>0.7926</u> $\pm$ .0014 | <u>0.5212</u> $\pm$ .0005 |
| Infograph      | ProS <sup>†</sup> [8]          | CVPR'24  | 0.5634                    | 0.5229                    | 0.5065                    | 0.4745                    | 0.6152                    | 0.3965                    |
|                | UCDR-Adapter <sup>†</sup> [14] | WACV'25  | 0.5803                    | 0.5476                    | 0.5359                    | 0.5093                    | 0.6257                    | 0.4086                    |
|                | DeFT-LoRA [36]                 | AAAI'26  | 0.6093                    | 0.5735                    | <b>0.5521</b>             | <b>0.5242</b>             | 0.6454                    | 0.4171                    |
|                | DePro <sup>†</sup> [3]         | SIGIR'25 | 0.6174                    | 0.5895                    | 0.5416                    | 0.5195                    | <b>0.6820</b>             | <b>0.4369</b>             |
|                | CAPT-shallow                   | –        | <b>0.6201</b>             | <u>0.5901</u>             | <u>0.5436</u>             | 0.5204                    | 0.6691                    | 0.4247                    |
|                | <b>CAPT-deep (ours)</b>        | –        | 0.6200 $\pm$ .0017        | <b>0.5928</b> $\pm$ .0014 | 0.5422 $\pm$ .0016        | 0.5230 $\pm$ .0016        | 0.6740 $\pm$ .0049        | 0.4360 $\pm$ .0029        |
| Painting       | ProS <sup>†</sup> [8]          | CVPR'24  | 0.7488                    | 0.6935                    | 0.7080                    | 0.6576                    | 0.7287                    | 0.4596                    |
|                | UCDR-Adapter <sup>†</sup> [14] | WACV'25  | 0.7547                    | 0.6986                    | 0.7216                    | 0.6706                    | 0.7385                    | 0.4724                    |
|                | DeFT-LoRA [36]                 | AAAI'26  | 0.7551                    | 0.6994                    | 0.7135                    | 0.6639                    | 0.7415                    | 0.4752                    |
|                | DePro <sup>†</sup> [3]         | SIGIR'25 | 0.7701                    | 0.7185                    | 0.7181                    | 0.6726                    | <b>0.7738</b>             | <b>0.4911</b>             |
|                | CAPT-shallow                   | –        | 0.7724                    | 0.7193                    | <b>0.7251</b>             | <b>0.6778</b>             | 0.7506                    | 0.4701                    |
|                | <b>CAPT-deep (ours)</b>        | –        | <b>0.7728</b> $\pm$ .0007 | <b>0.7223</b> $\pm$ .0003 | 0.7214 $\pm$ .0010        | <b>0.6778</b> $\pm$ .0006 | 0.7707 $\pm$ .0017        | 0.4859 $\pm$ .0007        |
| Quickdraw      | ProS <sup>†</sup> [8]          | CVPR'24  | 0.2718                    | 0.2456                    | 0.2230                    | 0.2049                    | 0.2660                    | 0.1148                    |
|                | UCDR-Adapter <sup>†</sup> [14] | WACV'25  | 0.2800                    | 0.2512                    | 0.2334                    | 0.2140                    | 0.2775                    | 0.1153                    |
|                | DeFT-LoRA [36]                 | AAAI'26  | 0.3012                    | 0.2668                    | 0.2522                    | 0.2279                    | 0.3010                    | 0.1194                    |
|                | DePro <sup>†</sup> [3]         | SIGIR'25 | 0.3195                    | 0.2979                    | 0.2438                    | 0.2343                    | <b>0.3359</b>             | <b>0.1352</b>             |
|                | CAPT-shallow                   | –        | 0.2984                    | 0.2712                    | 0.2287                    | 0.2116                    | 0.2904                    | 0.1191                    |
|                | <b>CAPT-deep (ours)</b>        | –        | <b>0.3278</b> $\pm$ .0055 | <b>0.3017</b> $\pm$ .0048 | <b>0.2551</b> $\pm$ .0028 | <b>0.2415</b> $\pm$ .0025 | 0.3183 $\pm$ .0056        | 0.1278 $\pm$ .0026        |
| <b>Average</b> | ProS <sup>†</sup> [8]          | CVPR'24  | 0.5979                    | 0.5550                    | 0.5470                    | 0.5110                    | 0.6345                    | 0.3991                    |
|                | UCDR-Adapter <sup>†</sup> [14] | WACV'25  | 0.6088                    | 0.5665                    | 0.5666                    | 0.5312                    | 0.6437                    | 0.4067                    |
|                | DeFT-LoRA [36]                 | AAAI'26  | 0.6212                    | 0.5776                    | 0.5690                    | 0.5331                    | 0.6547                    | 0.4105                    |
|                | DePro <sup>†</sup> [3]         | SIGIR'25 | 0.6364                    | 0.5996                    | 0.5686                    | 0.5398                    | <b>0.6881</b>             | <b>0.4282</b>             |
|                | CAPT-shallow                   | –        | 0.6310                    | 0.5913                    | 0.5664                    | 0.5345                    | 0.6616                    | 0.4092                    |
|                | <b>CAPT-deep (ours)</b>        | –        | <b>0.6428</b> $\pm$ .0010 | <b>0.6057</b> $\pm$ .0009 | <b>0.5753</b> $\pm$ .0008 | <b>0.5473</b> $\pm$ .0008 | 0.6829 $\pm$ .0008        | 0.4240 $\pm$ .0001        |

Table 2. Frozen-backbone comparison, DomainNet average (mAP@200 / Prec@200). DePro-Frozen removes LayerNorm tuning; DePro-LN matches Table 1. CAPT+LN(+bias): the same freedom for CAPT (three seeds). **Bold** = best.

| method       | Unseen             | Mixed              | UdCDR              |
|--------------|--------------------|--------------------|--------------------|
| DePro-Frozen | .6132/.5783        | .5426/.5163        | .6765/.4257        |
| DePro-LN     | .6364/.5996        | .5686/.5398        | <b>.6881/.4282</b> |
| <b>CAPT</b>  | .6428/.6057        | .5753/.5473        | .6829/.4240        |
| CAPT+LN      | .6455/.6081        | .5781/.5501        | .6853/.4248        |
| CAPT+LN+bias | <b>.6458/.6085</b> | <b>.5782/.5502</b> | .6877/.4260        |

mAP 0.6364  $\rightarrow$  0.6428, Mixed 0.5686  $\rightarrow$  0.5753) and DePro stays ahead on average UdCDR mAP (0.6881 vs. 0.6829): the advantage is consistent rather than large. With  $\sim$ 1.35M trainable parameters (vs. DePro and ProS 18.5M, UCDR-Adapter 2.37M, DeFT-LoRA 1.29M, Fig. 1), CAPT matches the most parameter-efficient prior method while being more accurate. We do not claim a parameter advantage, as the gains come from DSIP and PACE.

**Significance.** Over three seeds on the sketch holdout, the CAPT mean exceeds DePro by +0.0084 (Unseen) and +0.0073 (Mixed), roughly 3–4 $\times$  the seed standard deviation, so those gains are not seed noise, while

UdCDR-mAP stays within noise of DePro.

**Fair frozen-backbone comparison.** The DePro entry in Table 1 fine-tunes CLIP’s LayerNorm affines (its public “LN trick”), so its backbone is not strictly frozen. With the backbone fully frozen, matching our setting, DePro drops (Table 2) and CAPT wins 26 of the 30 cells, including all 20 Unseen and Mixed cells and the average UdCDR-mAP. The four cells CAPT still trails are all UdCDR. DePro’s UdCDR edge thus comes *entirely* from LayerNorm fine-tuning. The converse also holds: granting CAPT the same freedom (LN+bias, +0.2M; 1.55M total vs. DePro’s 18.5M) improves all six metrics (paired  $p \leq 0.031$ , fifteen runs) and closes the UdCDR-mAP gap to 0.0004 (0.6877 vs. 0.6881), well inside the seed deviation, while widening the Unseen and Mixed leads (Table 2).

#### 4.4. Cross-Dataset Generalization

To test transfer beyond DomainNet, we evaluate on the cross-dataset benchmarks Sketchy and TU-Berlin (sketch-to-image retrieval, unseen classes; Table 3). For a controlled comparison we reimplement ProS, DePro, and UCDR-Adapter in our environ-

Table 3. Cross-dataset generalization (sketch-to-image, unseen classes).  $\dagger$  = our same-environment reimplement (frozen CLIP ViT-B/32). Sketchy: full extended gallery, mAP@200 / Prec@200, as in SnMp-Net [25]/SASA [33]/ProS [8]; TU-Berlin: mAP@all / Prec@100. **Bold** = best; one run per entry (CAPT TU-Berlin: three-seed mean,  $\sigma$  0.0024/0.0022; DePro: lr-sweep best).

| Method                       | Sketchy                | TU-Berlin              |
|------------------------------|------------------------|------------------------|
|                              | mAP@200 / Prec@200     | mAP@all / Prec@100     |
| ProS $^\dagger$ [8]          | 0.7151 / 0.6739        | 0.6138 / 0.7037        |
| DePro $^\dagger$             | 0.7515 / 0.7177        | 0.6629 / 0.7203        |
| UCDR-Adapter $^\dagger$ [14] | 0.6933 / 0.6490        | 0.6314 / 0.7099        |
| <b>CAPT (ours)</b>           | <b>0.7638 / 0.7291</b> | <b>0.6641 / 0.7305</b> |

Table 4. **Isolating the routing signal.** Each row replaces only the rule producing  $w_i$  (temperature-matched). Top: reported rule, absolute five-foldout mean; body: its advantage ( $10^{-3}$  mAP) over five holdouts  $\times$  three seeds (sign test, Holm over eight comparisons); learned schemes: sketch holdout, three seeds. Grey = tie (sign flips across benchmarks).

| routing rule replaced by                                                                       | advantage of the reported rule (> 0: ours better) |       |            | wins  | $P_{\text{Holm}}$ |
|------------------------------------------------------------------------------------------------|---------------------------------------------------|-------|------------|-------|-------------------|
|                                                                                                | Unseen                                            | Mixed | U $^d$ CDR |       |                   |
| reference: combined rule (absolute mAP)                                                        | .6428                                             | .5752 | .6839      | n/a   | n/a               |
| shuffled $w_i$ within batch                                                                    | +0.8                                              | +1.5  | +1.2       | 13/15 | 0.022             |
| fixed $w=0.3$                                                                                  | +0.5                                              | +0.8  | +2.6       | 14/15 | 0.006             |
| fixed $w=0.5$                                                                                  | +1.0                                              | +1.8  | +1.2       | 14/15 | 0.006             |
| fixed $w=0.7$                                                                                  | +2.8                                              | +3.9  | +0.6       | 15/15 | 0.0005            |
| fixed functions of the same statistics (within $\pm 0.002$ ; grey = statistical tie with ours) |                                                   |       |            |       |                   |
| domain gap only $\sigma(-\hat{g}_i)$                                                           | +0.9                                              | +1.7  | +1.3       | 14/15 | 0.006             |
| confidence only $\sigma(\hat{c}_i)$                                                            | -0.4                                              | -0.4  | -0.1       | 4/15  | 0.53 (n.s.)       |
| focal shaping [21]                                                                             | -1.3                                              | -1.7  | -0.4       | 5/15  | 0.53 (n.s.)       |
| uncertainty wt. [15]                                                                           | +11.7                                             | +12.0 | +2.2       | 15/15 | 0.0005            |

ment (frozen CLIP ViT-B/32, identical data and training). On the standard 100-image-per-class subset, our ProS reimplement matches its published Sketchy mAP@200 (0.6993 vs. 0.6991), validating our pipeline; the table reports the full-gallery protocol of SnMp-Net [25]/SASA [33]/ProS [8]. Under this protocol, CAPT leads *every* metric on both Sketchy and TU-Berlin, demonstrating strong performance across standard UCDR benchmarks.

#### 4.5. Analysis

We analyze CAPT by decomposing the prompt generator, the alignment objective, and the refinement layer. All analysis is performed on the sketch holdout unless otherwise specified.

**Ablation on DSIP and PACE.** Table 5 summarizes the sketch-holdout ablation in a cumulative way. The anchored read-out, where static queries read prompts from the frozen text-class anchors, accounts for most of the gain over the frozen baseline (Unseen-mAP 0.39  $\rightarrow$  0.66). Image-conditioned anchors complete DSIP and give a small but consistent gain on all six metrics. Adding CDCM further improves Unseen and Mixed retrieval, but slightly reduces UdCDR precision, showing the contrastive-prototype conflict. PACE resolves this conflict by routing each sample between CDCM and CDPA, and gives the best UdCDR scores without

Table 5. CAPT build-up on the sketch holdout (seed 0; mAP@200 / Prec@200). Rows are added cumulatively from a frozen CLIP + ITC baseline. **Bold** = best. The build-up ran on an earlier code revision: its final row (.6766) and the Table 1 entry (.6944) are not a matched pair, and their difference is *not* the refinement layer (matched ablation reported separately).

| variant              | Unseen              | Mixed               | UdCDR               |
|----------------------|---------------------|---------------------|---------------------|
| Frozen + ITC         | .3901/.3105         | .3391/.2644         | .4468/.2530         |
| + read-out           | .6602/.6209         | .5867/.5539         | .7551/.4998         |
| + image-cond. (DSIP) | .6688/.6300         | .5880/.5586         | .7669/.5113         |
| + CDCM               | <b>.6770</b> /.6342 | <b>.5981</b> /.5656 | .7712/.5084         |
| + PACE (full)        | .6766/ <b>.6362</b> | .5962/ <b>.5665</b> | <b>.7729</b> /.5131 |

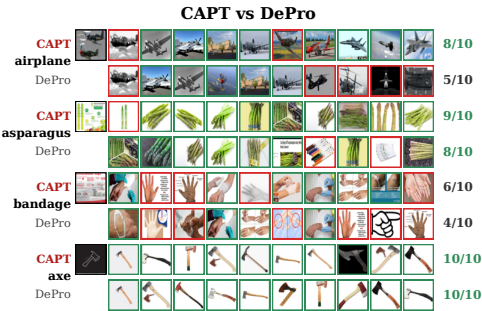


Figure 3. **Qualitative retrieval comparison.** Top-10 infograph-Unseen retrieval results for shared queries comparing CAPT with DePro.

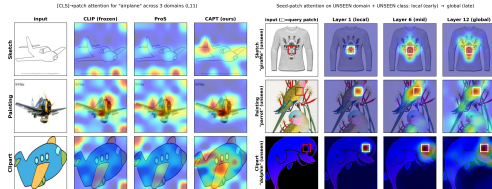


Figure 4. **Attention visualization.** Examples of CAPT final-layer attention maps on infograph-Unseen queries.

an extra loss-weight search. We also tried a per-image query modulated by the CLS token, but it gives no further gain (Unseen-mAP 0.6688  $\rightarrow$  0.6683), so CAPT keeps the static query.

**Qualitative results.** Figure 3 compares CAPT and DePro on shared infograph-Unseen queries. CAPT retrieves more correct top-10 results for several queries, improving airplane from 5/10 to 8/10, asparagus from 8/10 to 9/10, and bandage from 4/10 to 6/10, while matching DePro on the easier axe case. This is consistent with the quantitative gains and shows that CAPT returns more reliable same-class neighbors under large domain shifts. Figure 4 further shows that CAPT's final-layer attention is mainly concentrated on object regions rather than the surrounding background. Together, the retrieval examples and attention maps suggest that CAPT improves UCDR by preserving class-relevant object cues across different visual styles.

**Refined prototype bank.** The refinement layer

(Sec. 3) also feeds PACE: the same refined feature is written into PACE’s memory bank, so PACE routes samples using prototypes built in the same feature space as the final retrieval embedding. The adapter is not a contribution, but it matters for the aggregate: in a matched re-evaluation (same code and evaluation path, five holdouts; supp.), removing it moves the tally against DePro from 22/30 to 14/30, mostly on info-graph (+0.0224 Unseen-mAP) and sketch (+0.0098). It is necessary for the lead but not its source: head-only reaches 0.6217 on sketch (DePro 0.6843), and grafted onto DePro it moves  $\leq 0.0036$ .

**Design ablations.** Table 6 changes one component of the full model at a time (sketch holdout). (i) *Prompt read-out.* Replacing the MHA read-out in Eq. 2 with a plain linear projection lowers every metric (Unseen-mAP 0.6944  $\rightarrow$  0.6899): attending over the anchors is what turns them into a useful prompt. Against the free-prompt control (Table 6), full DSIP improves Unseen-mAP by 0.0072 ( $p=0.0009$ ; 12/15 matched runs over all five holdouts), with image conditioning accounting for 0.0054 ( $p=0.0027$ ): a structured read-out beats directly learning unconstrained prompt values. (ii) *Anchor source.* A random basis (0.6923) or vision-only patch tokens (0.6903) do not improve over the text anchors. The margin over a random basis is small (0.0021 Unseen-mAP): DSIP’s gain comes from the structured, image-conditioned read-out in (i), while the text anchors add a consistent but modest CLIP-aligned prior and keep the best Unseen and Mixed scores. The constraint is tight: the prompts’ residual onto the conditioned anchor hull is  $< 2 \times 10^{-7}$  at all 12 layers (patch tokens  $\approx 1.00$ , random 1.34–1.41). (iii) *PACE routing weight.* A fixed  $w=0.5$  trails per-sample routing on Unseen and Mixed (0.6915 vs. 0.6944), gaining marginally on UdCDR (0.7958); the constant-weight trade-off Table 4 isolates. Together, these ablations support the structured anchor-conditioned read-out, instance-specific image conditioning, a modest anchor prior, and per-sample routing.

**Routing.** Table 4 isolates the routing signal. No constant weight is competitive on DomainNet or either cross-dataset benchmark (10/10 runs,  $p=0.002$ ), and shuffling  $w_i$  within the batch costs as much as freezing it: the per-sample assignment carries the information. Confidence alone falls behind on every TU-Berlin metric, the gap alone to last on Sketchy; the combined rule of Eq. 7 is never worse than second. Stated precisely: over five holdouts and three seeds, PACE’s Unseen gain is +0.0026 (n.s.,  $p=0.19$ ) while its UdCDR-mAP gain is +0.0077 ( $p=0.005$ ), and confidence-only routing is indistinguishable from the combined rule; PACE’s contribution is a removed loss weight and Ud-

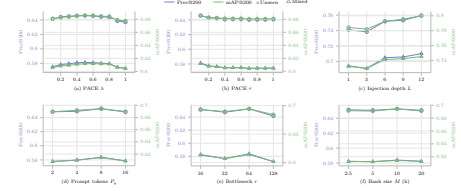


Figure 5. **Sensitivity analysis.**  $\lambda$  (a),  $\tau$  (b), depth  $L$  (c, clipart),  $P_q$  (d),  $r$  (e), bank  $M$  (f); sketch holdout unless noted. Circles/triangles = Unseen/Mixed; purple/green = Prec@200/mAP@200. Defaults sit at or next to the optimum.

Table 6. Design ablations on the full CAPT model (sketch holdout, seed 0, mAP@200 / Prec@200). Each block changes one component; the last row is CAPT. **Bold** = best per column. Routing: Table 4.

| variant                                | Unseen             | Mixed              | UdCDR              |
|----------------------------------------|--------------------|--------------------|--------------------|
| <i>Prompt read-out</i>                 |                    |                    |                    |
| linear (no MHA)                        | .6899/.6483        | .6092/.5792        | .7897/.5191        |
| free prompts (VPT-Deep, 0.07M)         | .6854/.6450        | .6044/.5752        | .7906/.5183        |
| <i>Anchor source</i>                   |                    |                    |                    |
| random basis                           | .6923/.6508        | .6118/.5815        | .7897/.5211        |
| vision-only                            | .6903/.6481        | .6090/.5779        | <b>.7954/.5256</b> |
| CAPT (MHA, semantic text anchors, gap) | <b>.6944/.6527</b> | <b>.6141/.5841</b> | .7926/.5212        |

CDR protection, not a higher Unseen score. Learned weights degenerate as pre-registered (an MLP [31] converges to  $w \equiv 1.0$ ; uncertainty weighting [15] loses all fifteen runs,  $p_{\text{Holm}}=0.0005$ ; GroupDRO [28] collapses onto one domain), while every fixed function of stop-gradient statistics, focal shaping [21] included, stays within 0.002 of the reported rule.

#### 4.6. Sensitivity Analysis

In Figure 5,  $\lambda$  peaks stably in  $[0.5, 0.6]$ ,  $\tau$  is best at its smallest value, accuracy rises with injection depth up to the full  $L=12$ , and  $P_q$ ,  $r$ ,  $M$  are flat around the defaults: no per-domain retuning is needed.

**Backbone scale.** On ViT-B/16 both methods improve and the ranking follows the stopping rule: own schedules 1/6 cells for CAPT, matched epoch 1 3/6, matched epoch 4 5/6; the backbone helps, but we do not claim the margin transfers.

**Limitations.** DePro with LayerNorm tuning stays slightly stronger on a few UdCDR precision cells, and PACE’s prototypes degrade under severe imbalance. Extending CAPT beyond image-to-image retrieval is future work.

## 5. Conclusion

CAPT generates a text-anchored, instance-conditional deep prompt per image and resolves the contrastive-prototype conflict per sample with a data-derived weight, using only  $\sim 1.35\text{M}$  parameters on a frozen CLIP.

## References

- [1] Ayan Kumar Bhunia, Pinaki Nath Chowdhury, Aneeshan Sain, Yongxin Yang, Tao Xiang, and Yi-Zhe Song. More photos are all you need: Semi-supervised learning for fine-grained sketch based image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4247–4256, 2021. 1
- [2] Abhra Chaudhuri, Ayan Kumar Bhunia, Yi-Zhe Song, and Anjan Dutta. Data-free sketch-based image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12084–12093, 2023. 1
- [3] Kaixiang Chen, Pengfei Fang, and Hui Xue. Depro: Domain ensemble using decoupled prompts for universal cross-domain retrieval. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2025. 1, 3, 5, 6
- [4] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 794–803, 2018. 3
- [5] Ayush Chopra, Abhishek Sinha, Hitesh Gupta, Mausoom Sarkar, Kumar Ayush, and Balaji Krishnamurthy. Powering robust fashion retrieval with information rich feature embeddings. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 326–334, 2019. 1
- [6] Sounak Dey, Pau Riba, Anjan Dutta, Josep Lladós, and Yi-Zhe Song. Doodle to search: Practical zero-shot sketch-based image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1
- [7] Anjan Dutta and Zeynep Akata. Semantically tied paired cycle consistency for zero-shot sketch-based image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1
- [8] Kaipeng Fang, Jingkuan Song, Lianli Gao, Pengpeng Zeng, Zhi-Qi Cheng, Xiyao Li, and Heng Tao Shen. Pros: Prompting-to-simulate generalized knowledge for universal cross-domain retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17292–17301, 2024. 1, 3, 5, 6, 7
- [9] Bojana Gajić and Ramón Baldrich. Cross-domain fashion image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018. 1
- [10] 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. *International Journal of Computer Vision (IJCV)*, 2024. 2
- [11] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9729–9738, 2020. 3
- [12] Xinwei He, Yang Zhou, Zhichao Zhou, Song Bai, and Xiang Bai. Triplet-center loss for multi-view 3d object retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1945–1954, 2018. 3
- [13] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 709–727, 2022. 1, 2
- [14] Haoyu Jiang, Zhi-Qi Cheng, Gabriel Moreira, Jiawen Zhu, Jingdong Sun, Bukun Ren, Jun-Yan He, Qi Dai, and Xian-Sheng Hua. Ucdr-adapter: Exploring adaptation of pre-trained vision-language models for universal cross-domain retrieval. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025. 1, 3, 5, 6, 7
- [15] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7482–7491, 2018. 3, 7, 8
- [16] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19113–19122, 2023. 1, 2
- [17] Muhammad Uzair Khattak, Syed Talal Wasim, Muza-mmil Naseer, Salman Khan, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Self-regulating prompts: Foundational model adaptation without forgetting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15190–15200, 2023. 2
- [18] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 18661–18673, 2020. 3
- [19] Minh Le, Anh Nguyen, Huy Nguyen, Chau Nguyen, Anh Tran, and Nhat Ho. Revisit visual prompt tuning: The expressiveness of prompt experts. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2026. Introduces VAPT (Visual Adaptive Prompt Tuning); arXiv:2501.18936. 2
- [20] Junnan Li, Pan Zhou, Caiming Xiong, and Steven C.H. Hoi. Prototypical contrastive learning of unsupervised representations. In *International Conference on Learning Representations (ICLR)*, 2021. 3
- [21] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 7, 8
- [22] Jochem Loedeman, Maarten C. Stol, Tengda Han, and Yuki M. Asano. Prompt generation networks for input-

- space adaptation of frozen vision transformers. In *British Machine Vision Conference (BMVC)*, 2024. 2
- [23] Haodong Lu, Xinyu Zhang, Kristen Moore, Jason Xue, Lina Yao, Anton van den Hengel, and Dong Gong. Continual learning on CLIP via incremental prompt tuning with intrinsic textual anchors. *arXiv preprint arXiv:2505.20680*, 2025. 2
- [24] Massimiliano Mancini, Zeynep Akata, Elisa Ricci, and Barbara Caputo. Towards recognizing unseen categories in unseen domains. In *European Conference on Computer Vision (ECCV)*, 2020. 1, 5
- [25] Soumava Paul, Titir Dutta, and Soma Biswas. Universal cross-domain retrieval: Generalizing across classes and domains. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12056–12064, 2021. 1, 3, 5, 7
- [26] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 8748–8763, 2021. 1
- [27] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 3
- [28] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *ICLR*, 2020. 3, 8
- [29] Aneeshan Sain, Ayan Kumar Bhunia, Pinaki Nath Chowdhury, Subhadeep Koley, Tao Xiang, and Yi-Zhe Song. CLIP for all things zero-shot sketch-based image retrieval, fine-grained or not. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3
- [30] Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. The sketchy database: Learning to retrieve badly drawn bunnies. *ACM Transactions on Graphics (TOG)*, 2016. 1
- [31] Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. In *NeurIPS*, 2019. 8
- [32] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 14274–14289, 2022. 2
- [33] Jialin Tian, Xing Xu, Kai Wang, Zuo Cao, Xunliang Cai, and Heng Tao Shen. Structure-aware semantic-aligned network for universal cross-domain retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 278–289, 2022. 7
- [34] Lixu Wang, Xinyu Du, and Qi Zhu. Semantic feature learning for universal unsupervised cross-domain retrieval. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. Introduces U<sup>2</sup>CDR with an instance-prototype-mixed contrastive loss; arXiv:2403.05690. 3
- [35] Yinghui Xing, Qirui Wu, De Cheng, Shizhou Zhang, Guoqiang Liang, Peng Wang, and Yanning Zhang. Dual modality prompt tuning for vision-language pre-trained model. *IEEE Transactions on Multimedia (TMM)*, 2023. Proposes DPT and the class-aware visual prompt (CAVPT). 2
- [36] Ke Xu, Xiaozheng Shen, Shanshan Wang, Mengzhu Wang, and Xun Yang. Deft-lora: Decoupled and fused tuning with lora experts for universal cross-domain retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 11296–11304, 2026. 1, 3, 5, 6
- [37] Hantao Yao, Rui Zhang, and Changsheng Xu. Visual-language prompt tuning with knowledge-guided context optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6757–6767, 2023. 2
- [38] Renrui Zhang, Wei Zhang, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free adaption of CLIP for few-shot classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2
- [39] Hao Zheng, Shunzhi Yang, Zhuoxin He, Jinfeng Yang, and Zhenhua Huang. Hierarchical cross-modal prompt learning for vision-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. 2
- [40] Kaiyang Zhou, Jingkan 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 (CVPR)*, pages 16816–16825, 2022. 2
- [41] Kaiyang Zhou, Jingkan Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision (IJCV)*, 130(9):2337–2348, 2022. 1
- [42] Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15659–15669, 2023. 2