CASPA: Graph-Structured Concept Anchors for Modality-Agnostic Adaptation in Vision-Language Models

Supplementary Material

S1. Supplementary Overview

We follow the evaluation protocols established in various prior works [3, 6, 22]. To assess CASPA's task-agnostic capabilities, we adopt multiple experimental setups and compare it with a range of state-of-the-art methods [3, 6, 8, 13, 16, 17, 21, 22, 24, 31, 40–43, 46–50] across different benchmarks. The supplementary material is organized as follows:

1. **Methodological Justifications and Complexity Summaries in Subsection S1.1**
   - More details on the methodological reasoning behind CASPA.
   - Provide summary of computational complexity, such as parameters, FLOPs, training time in Table 8.
   - Present step-by-step pseudo-codes (Algorithms 1 and 2) for CASPA.
2. **Experimental Setup and Datasets Used in Subsection S1.2**
   - Outline the experimental protocols for reproducibility.
   - Offer detailed descriptions (Table 9) of all datasets.
3. **Additional plug-and-play benchmarking with SOTA Methods in Subsection S1.3**
   - Compare CASPA + CoOp against vanilla CoOp (popularly used for comparison) [49] and a recent two-stage state-of-the-art adaptation method [6].
4. **Ablation Studies on ImageNet [5] in S1.3**
   - Provide ablation results for CASPA + CoOp (Table 13) and show that CASPA reduces generalization gaps caused by source domain overfitting, even as the number of shots increases.
5. **Qualitative Visualizations in S1.3**
   - Include examples highlighting CASPA's zero-shot recognition ability (Figure 7) by showing performance on most confusing samples where CLIP struggles.
6. **Discussion on Advantages and Limitations in Subsection S1.4**
   - Conclude with a discussion of CASPA's strengths and its potential limitations.

S1.1. CASPA: Conceptual Overview

CASPA is designed around the idea that the downstream adaptation of vision–language models [14, 19, 31, 35, 37] should exploit the *shared conceptual manifold* already encoded in the pretrained CLIP embedding space. Rather than optimizing a separate learnable prompt for each class—an approach commonly used in prompt-tuning methods, CASPA introduces a compact set of *semantic anchors* that serve as global basis elements. These anchors are modality-specific, as defined in (Eq. 2), and also rewritten below:

$$\mathcal{A}_m = \{\mathbf{a}_m^{(k)} \in \mathbb{R}^d \mid k = 1, \ldots, K_m\}, \quad m \in \{t, v\},$$

and they are shared across all classes. Classes do not learn independent textual vectors; instead, each class forms a mixture over anchors via association coefficients (Eq. 3), and rewritten as

$$\boldsymbol{\pi}_m^{(c)} \in \Delta^{K_m - 1},$$

which produce adapted prototypes (Eq. 5), and is rewritten as

$$\mathbf{z}_m^{(c)} = \mathrm{Norm}\left(\mathbf{b}_m^{(c)} + \sum_{k=1}^{K_m} \pi_{m,k}^{(c)} \mathbf{a}_m^{(k)} + \mathbf{s}_m^{(c)}\right).$$

Let $\mathbf{A}_m = [\mathbf{a}_m^{(1)}, \ldots, \mathbf{a}_m^{(K_m)}] \in \mathbb{R}^{d \times K_m}$ denote the anchor matrix. We equivalently write the anchor mixture as $\mathbf{A}_m \boldsymbol{\pi}_m^{(c)} \equiv \sum_{k=1}^{K_m} \pi_{m,k}^{(c)} \mathbf{a}_m^{(k)} = \mathbf{M}_m^{(c)}$, consistent with the notation used in the main paper. These prototypes unify three components: (i) the frozen CLIP embedding $\mathbf{b}_m^{(c)}$, (ii) the semantic anchor mixture, and (iii) a lightweight residual $\mathbf{s}_m^{(c)}$ that locally refines class-specific patterns. This induces a *shared semantic space* that is smooth, compositional, and aligned across modalities, and provides the foundation for the two major design principles detailed below.

Justification I: Parameter Sharing in CASPA. Compared to methods like CoOp [49], CASPA enjoys substantially better parameter scaling. In class-specific CoOp, each class learns M_{ctx} *context tokens*, each with a d-dimensional vector, leading to a total parameter cost of $\mathcal{O}(CM_{ctx}d)$, where C is the number of classes. This linear growth in C makes CoOp increasingly expensive.

In CASPA, the situation is different. The model uses K_t *text anchors* and K_v *image anchors*, each of dimension d, which are *shared across all classes* and thus incur only a constant cost independent of C. For each class, CASPA learns only: (1) a (K_t+K_v)-dimensional *association vector* that mixes the anchors, and (2) two d-dimensional *residual shifts* that refine the class embedding. The resulting parameter complexity is

$$\underbrace{\mathcal{O}(K_t d + K_v d)}_{\text{anchors}} + \underbrace{\mathcal{O}(C(K_t + K_v))}_{\text{associations}} + \underbrace{\mathcal{O}(Cd)}_{\text{residuals}},$$

where K_t and K_v are typically small, making CASPA more scalable in C. Table 8 provides a summary of CASPA's resource usage on ImageNet. The model is lightweight and

efficient, with 1.1M parameters and low GFLOPs per sample ($\approx$ 0.07 GFLOPs), along with low memory consumption and competitive training time.

Justification II: Semantic Factorization and Reconstruction. CASPA is built on the assumption that downstream class semantics occupy a *low-dimensional, shared conceptual subspace* embedded within the CLIP joint manifold. To formalize this, let the adapted textual prototypes be collected as

$$\mathbf{Z}_t = \begin{bmatrix} (\mathbf{z}_t^{(1)})^\top \\ \vdots \\ (\mathbf{z}_t^{(C)})^\top \end{bmatrix} \in \mathbb{R}^{C\times d},$$

and define the corresponding Base embeddings as

$$\mathbf{B}_t = \begin{bmatrix} (\mathbf{b}_t^{(1)})^\top \\ \vdots \\ (\mathbf{b}_t^{(C)})^\top \end{bmatrix} \in \mathbb{R}^{C\times d}.$$

We further define the textual anchor bank as

$$\mathbf{A}_t = \begin{bmatrix} \mathbf{a}_t^{(1)} & \cdots & \mathbf{a}_t^{(K_t)} \end{bmatrix} \in \mathbb{R}^{d\times K_t}.$$

From the CASPA formulation (Eq. 5), each prototype is produced as

$$\mathbf{z}_t^{(c)} = \mathrm{Norm}\Big(\mathbf{b}_t^{(c)} + \mathbf{A}_t\boldsymbol{\pi}_t^{(c)} + \mathbf{s}_t^{(c)}\Big).$$

When the learned residuals $\mathbf{s}_t^{(c)}$ remain small, as encouraged by anchor sharing, each prototype becomes well-approximated (up to a small residual) by the normalized combination $\mathrm{Norm}(\mathbf{b}_t^{(c)}+\mathbf{A}_t\boldsymbol{\pi}_t^{(c)})$. Stacking terms over all C classes, therefore, yields the approximate factorization

$$\mathbf{Z}_t \approx \mathbf{B}_t+\boldsymbol{\Pi}_t\mathbf{A}_t^\top \quad\Longleftrightarrow\quad \mathbf{z}_t^{(c)} \approx \mathrm{Norm}\Big(\mathbf{b}_t^{(c)} + \mathbf{A}_t\boldsymbol{\pi}_t^{(c)}\Big), \tag{16}$$

where $\boldsymbol{\Pi}_t$ stacks all association vectors.

The reconstruction quality is quantified by the Frobenius deviation

$$\|\mathbf{Z}_t - \mathbf{B}_t - \boldsymbol{\Pi}_t\mathbf{A}_t^\top\|_F = \left(\sum_c \|\mathbf{z}_t^{(c)} - \mathbf{b}_t^{(c)} - \mathbf{A}_t\boldsymbol{\pi}_t^{(c)}\|_2^2\right)^{1/2}. \tag{17}$$

If this deviation is small, then the anchors $\mathcal{A}_t$ span a *shared conceptual basis* for the dataset's semantics, meaning that a small K_t suffices to approximate the (potentially high-dimensional) class structure.

Lemma 1. *Let*

$$\mathbf{Z}_t \in \mathbb{R}^{C\times d}, \qquad \mathbf{A}_t \in \mathbb{R}^{d\times K_t}, \qquad \boldsymbol{\Pi}_t \in \mathbb{R}^{C\times K_t},$$

and define the residual matrix as

$$\mathbf{E}(\mathbf{Z}_t - \mathbf{B}_t) - \boldsymbol{\Pi}_t\mathbf{A}_t^\top.$$

If $\|\mathbf{E}\|_F \le \varepsilon$, then every prototype lies within the Euclidean distance at most ε from the anchor subspace:

$$\forall c \in \{1,\dots,C\}: \qquad \mathrm{dist}\big(\mathbf{z}_t^{(c)}, \mathrm{span}(\mathbf{A}_t)\big) \le \varepsilon.$$

Thus, small Frobenius deviation guarantees that the anchors form an (approximate) shared semantic basis for all classes. Here, ε denotes an upper bound on the Frobenius reconstruction error.

Proof. We write $\mathbf{E} = [\mathbf{e}_1^\top;\dots;\mathbf{e}_C^\top]$ so that $\mathbf{e}_c = (\mathbf{z}_t^{(c)} - \mathbf{b}_t^{(c)}) - \mathbf{A}_t\boldsymbol{\pi}_t^{(c)}$. The Frobenius norm decomposes row-wise into a sum of squared Euclidean error over classes, since it is the ℓ_2 aggregation of all row residuals. Then

$$\|\mathbf{E}\|_F^2 = \sum_{c=1}^{C} \|\mathbf{e}_c\|_2^2 \le \varepsilon^2.$$

Since each term is non-negative, every row satisfies $\|\mathbf{e}_c\|_2 \le \varepsilon$. Because $\mathbf{A}_t\boldsymbol{\pi}_t^{(c)}$ lies in $\mathrm{span}(\mathbf{A}_t)$, the distance bound follows:

$$\mathrm{dist}\big(\mathbf{z}_t^{(c)}, \mathrm{span}(\mathbf{A}_t)\big) \le \|\mathbf{z}_t^{(c)} - \mathbf{A}_t\boldsymbol{\pi}_t^{(c)}\|_2 = \|\mathbf{e}_c\|_2 \le \varepsilon.$$

□

Discussion. (Eq. 16) shows that CASPA effectively seeks a low-rank approximation of the adapted textual prototypes using a shared anchor dictionary $\mathbf{A}_t$ and class-specific mixing coefficients $\boldsymbol{\Pi}_t$. By formalizing the reconstruction error through the Frobenius norm, (Eq. 17), the lemma establishes a direct geometric interpretation: small global reconstruction error implies small per-class distance from each prototype to the anchor subspace.

The discussion stated above utilizes two facts: (i) the Frobenius norm is the square root of the sum of squared row norms, and (ii) each anchor reconstruction $\mathbf{A}_t\boldsymbol{\pi}_t^{(c)}$ lies in $\mathrm{span}(\mathbf{A}_t)$. From $\|\mathbf{E}\|_F \le \varepsilon$, it follows that *every* prototype lies in an ε-neighborhood of the subspace. Thus, the lemma formalizes that CASPA does not merely compress the class representations, it ensures a uniform upper bound on how far any individual class embedding can drift from the shared semantic basis. Conceptually, this validates the central claim: if the learned anchors capture the core semantic directions of the dataset, then the class-specific components need only fine-tuning within a small neighborhood, keeping the adaptation stable, data-efficient, and resistant to overfitting. This provides theoretical support for the empirical observation that small anchor banks ($K_t \ll d$) suffice to model high-dimensional class semantics across diverse datasets. For clarity, we present the factorization on

the text side; the image-side derivation is entirely analogous by replacing $(\mathbf{Z}_t, \mathbf{A}_t, \mathbf{\Pi}_t)$ with their visual counterparts $(\mathbf{Z}_v, \mathbf{A}_v, \mathbf{\Pi}_v)$.

Remark. The lemma is purely conditional: it assumes only the reconstruction bound $\|\mathbf{E}\|_F \leq \varepsilon$ and does not require $\mathbf{E}$ to lie in $\mathrm{span}(\mathbf{A}_t)$. When prototypes are ℓ_2–normalized in practice, this ℓ_2 distance bound directly implies a small angular deviation, and ensures that the geometric interpretation remains valid.

This factorization perspective yields several emergent behaviors:

- **Compositionality.** New or rare categories are represented as mixtures of anchors, enabling a more robust zero-shot transfer.
- **Multi-class consistency.** Semantically related classes acquire similar mixture weights $\boldsymbol{\pi}_t^{(c)}$, promoting smoothness over the class manifold.
- **Geometric regularization.** Since $\boldsymbol{\pi}_t^{(c)} \in \Delta^{K_t-1}$, the anchor component $\mathbf{A}_t\boldsymbol{\pi}_t^{(c)}$ lies in the convex hull of anchors, constraining updates to a bounded region of the embedding space and promotes stable, non-degenerate directions.

Anchor quality is further enforced by the diversity regularizer (Eq. 6), and is rewritten as

$$\mathcal{L}_{\text{div}} = \sum_{m\in\{t,v\}} \left\|\mathbf{A}_m^\top \mathbf{A}_m - \mathbf{I}_{K_m}\right\|_F^2,$$

which encourages anchors to be approximately orthogonal, preventing "anchor collapse" and improving the effective rank of the conceptual basis.

Finally, the cross-modal alignment term (Eq. 9):

$$\mathcal{L}_{\text{xcr}} = \frac{1}{C}\sum_{c=1}^{C}\left(1 - \cos(\mathbf{M}_t^{(c)}, \mathbf{M}_v^{(c)})\right)$$

ensures that both modalities employ the same conceptual coordinate system. Together, these properties imply that CASPA performs a *regularized, multimodal matrix factorization* of class semantics that underpin its transferability.

Table 8. CASPA resource consumption on ImageNet

Metric	Value
Total Parameters	1.1 M
FLOPs per forward pass	0.07 GFLOPs
Forward pass memory	4.09 MB
Peak memory requirement	1501 MB
Total time	9 mins (A6000)

S1.2. Experimental Setup

Datasets and Experimental Setup. We evaluate CASPA across eleven widely adopted visual recognition benchmarks (Table 9) spanning generic object recognition (ImageNet, Caltech), fine-grained categorization (OxfordPets, StanfordCars, Flowers102, Food101, FGVC Aircraft), scene and texture understanding (SUN397, DTD), satellite imagery (EuroSAT), and human action recognition (UCF). These datasets cover a broad range of visual domains, class cardinalities, and sample sizes, and provide a comprehensive test bed for measuring transferability, few-shot adapta-

Algorithm 1 CASPA: Few-Shot Adaptation Pipeline

Require: Pretrained encoders f_θ, g_Φ, Base embeddings $\{\mathbf{b}_t^{(c)}, \mathbf{b}_v^{(c)}\}_{c=1}^C$, dataset $\{(x_i, y_i)\}_{i=1}^N$, anchors K_t, K_v, hyperparameters λ_d, λ_x, batch size B

Initialize anchors $\mathbf{A}_t \in \mathbb{R}^{d\times K_t}$, $\mathbf{A}_v \in \mathbb{R}^{d\times K_v}$
Initialize residuals $\{\mathbf{s}_t^{(c)}\}, \{\mathbf{s}_v^{(c)}\} \leftarrow 0$
Initialize learnable logit scale s
for each training step **do**
  Extract normalized image features:
  $\mathbf{X} \leftarrow [\mathrm{Norm}(f_\theta(x_1)), \ldots, \mathrm{Norm}(f_\theta(x_B))]^\top$
  Compute anchor association weights:
  for $c = 1 \ldots C$ **do**
    $\boldsymbol{\pi}_t^{(c)} \leftarrow \mathrm{softmax}(\mathbf{A}_t^\top \mathbf{b}_t^{(c)})$
    $\boldsymbol{\pi}_v^{(c)} \leftarrow \mathrm{softmax}(\mathbf{A}_v^\top \mathbf{b}_v^{(c)})$
  end for
  Compute anchor mixtures:
  for $c = 1 \ldots C$ **do**
    $\mathbf{M}_t^{(c)} \leftarrow \mathbf{A}_t\boldsymbol{\pi}_t^{(c)}$
    $\mathbf{M}_v^{(c)} \leftarrow \mathbf{A}_v\boldsymbol{\pi}_v^{(c)}$
  end for
  Compose adapted prototypes:
  for $c = 1 \ldots C$ **do**
    $\mathbf{z}_t^{(c)} \leftarrow \mathrm{Norm}\left(\mathbf{b}_t^{(c)} + \mathbf{M}_t^{(c)} + \mathbf{s}_t^{(c)}\right)$
    $\mathbf{z}_v^{(c)} \leftarrow \mathrm{Norm}\left(\mathbf{b}_v^{(c)} + \mathbf{M}_v^{(c)} + \mathbf{s}_v^{(c)}\right)$
  end for
  Form prototype matrix:
  $\mathbf{Z}_t \leftarrow [\mathbf{z}_t^{(1)}, \ldots, \mathbf{z}_t^{(C)}]^\top$
  Compute logits:
  $\mathbf{L} \leftarrow s \cdot \mathbf{X}\mathbf{Z}_t^\top$
  Compute losses:
  $\mathcal{L}_{\text{CE}} = -\frac{1}{B}\sum_{i=1}^B \log \frac{\exp(\mathbf{L}_{i,y_i})}{\sum_{c=1}^C \exp(\mathbf{L}_{i,c})}$
  $\mathcal{L}_{\text{xcr}} = \frac{1}{C}\sum_{c=1}^C \left(1 - \cos\left(\mathbf{M}_t^{(c)}, \mathbf{M}_v^{(c)}\right)\right)$
  $\mathcal{L}_{\text{div}} = \sum_{m\in\{t,v\}} \|\mathbf{A}_m^\top \mathbf{A}_m - \mathbf{I}_{K_m}\|_F^2$
  Total loss:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda_x \mathcal{L}_{\text{xcr}} + \lambda_d \mathcal{L}_{\text{div}}$$

  Update parameters:
  Update $\mathbf{A}_t, \mathbf{A}_v, \{\mathbf{s}_t^{(c)}\}, \{\mathbf{s}_v^{(c)}\}, s$ via gradient descent
end for

Table 9. Summary of datasets used for our experimentation. A detailed decription is also provided.

Dataset	Classes	Train	Validation	Test	Description	Prompt
ImageNet [5]	1000	1.28M	-	50k	Recognition of generic objects	"a photo of a [CLASS]."
Caltech [7]	100	4128	1649	2465	Recognition of generic objects	"a photo of a [CLASS]."
Oxford Pets [29]	37	2944	736	3669	Fine-grained pet classification	"a photo of a [CLASS], a type of pet."
Stanford Cars [20]	196	6509	1635	8041	Fine-grained car classification	"a photo of a [CLASS]."
Flowers102 [28]	102	4093	1633	2463	Fine-grained flower classification	"a photo of a [CLASS], a type of flower."
Food [1]	101	50,500	20,200	30,300	Fine-grained food recognition	"a photo of [CLASS], a type of food."
FGVC Aircraft [27]	100	3334	3333	3333	Fine-grained aircraft classification	"a photo of a [CLASS], a type of aircraft."
SUN397 [39]	397	15,880	3970	19,850	Scene classification	"a photo of a [CLASS]."
DTD [4]	47	2820	1128	1692	Texture classification	"[CLASS] texture."
EuroSAT [11]	10	13,500	5400	8100	Land use & satellite imagery	"a centered satellite photo of [CLASS]."
UCF [36]	101	7639	1898	3783	Human action recognition	"a photo of a person doing [CLASS]."

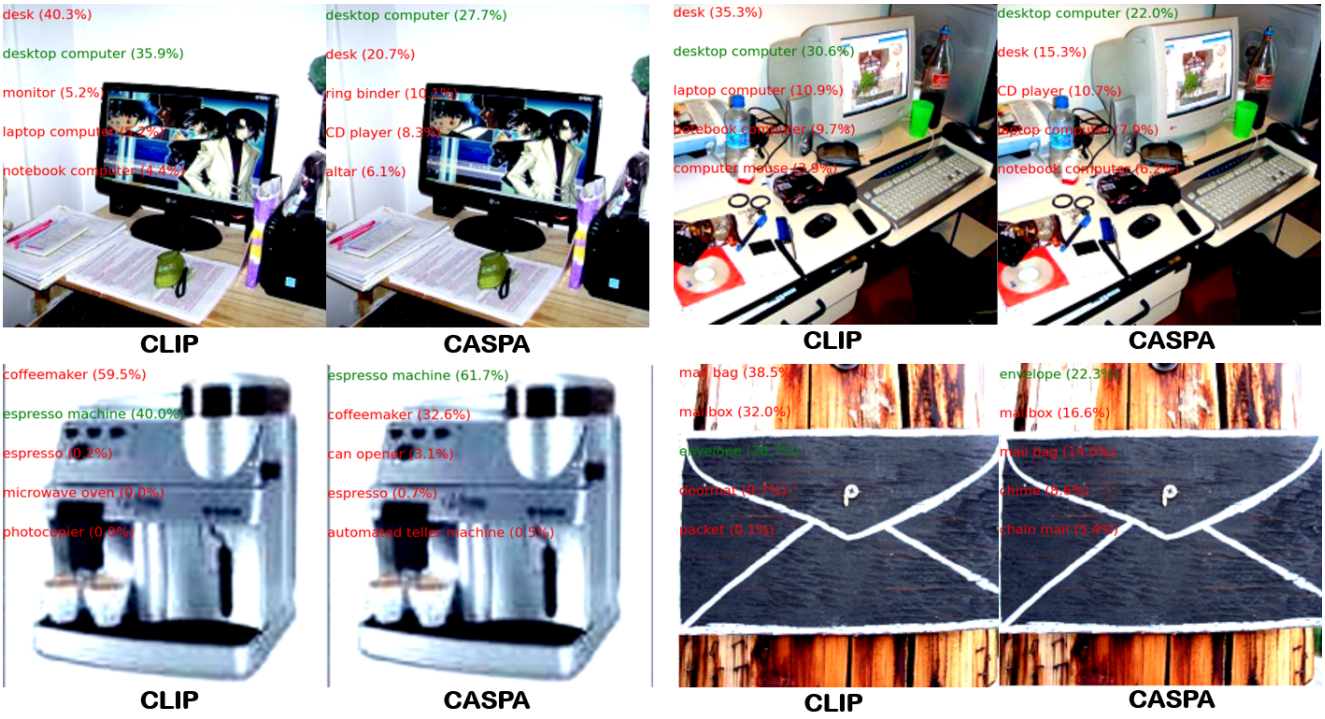

Figure 7. **Zero-shot predictions on confusing ImageNet Novel samples.** CLIP zero-shot predicts incorrectly, while CASPA correctly classifies the difficult Novel images with high confidence. For instance, the first image shows a desktop computer sitting on a desk. CLIP incorrectly predicts 'desk' because it over-relies on the dominant contextual object in the scene rather than the smaller, specific object (the computer). CASPA, correctly identifies the 'desktop computer' with 27.7% confidence by focusing on the object features rather than the surrounding context. The same is the case with the envelope and espresso machine images. Green denotes true label.

tion, and zero-shot capabilities of an adapter.

For all datasets, we use handcrafted prompt templates to construct text queries like prior methods [31, 46, 49]. All reported results in the main paper use ViT-B/16 as the default backbone. To maintain parity with prior methods, we consistently adopt this backbone across all experiments unless otherwise stated. All experiments employ the AdamW optimizer with an initial learning rate of 1×10^{-4} across all setups. For optimization, we use $K = 48$ learnable anchors, with a diversity regularization weight of $\lambda_{\text{d}} = 0.05$ and a consistency regularization weight of $\lambda_x = 150$. We adopt the `cosine` consistency mode throughout. A short warm-up phase is applied at the beginning of training for 10% of the total epochs. For ASAM, we set fixed $\rho = 0.1$ and $\eta = 0.01$ consistently across all experiments and datasets. The batch size is set to 4 for all experiment regimes, ex-

Algorithm 2 CASPA: Zero-Shot Adaptation to Novel Classes

Require: Learned textual anchors $\mathbf{A}_t = [\mathbf{a}_t^{(1)}, \ldots, \mathbf{a}_t^{(K_t)}] \in \mathbb{R}^{d \times K_t}$,

Base class embeddings $\{\mathbf{b}_t^{(c)}\}_{c=1}^{C}$, Novel class embedding $\mathbf{b}_t^{(\text{new})} \in \mathbb{R}^d$,

Normalized image feature $\mathbf{x} \in \mathbb{R}^d$, Confidence over base classes $\text{conf}_{\text{base}} \in [0, 1]$,

Scaling parameters $\alpha_{\text{min}}, \alpha_{\text{max}}$, Sharpness γ

Compute anchor association weights (vectorized softmax):

$$\boldsymbol{\pi}^{(\text{new})} \leftarrow \text{softmax}\big(\mathbf{A}_t^\top \mathbf{b}_t^{(\text{new})}\big) \in \mathbb{R}^{K_t}$$

▷ Softmax ensures convex combination over anchors

Compose adapted prototype via anchor mixing:

$$\mathbf{z}_t^{(\text{new})} \leftarrow \text{Norm}\Big(\mathbf{b}_t^{(\text{new})} + \mathbf{A}_t\, \boldsymbol{\pi}^{(\text{new})}\Big)$$

▷ Normalized prototype lies on unit hypersphere

Compute confidence-adaptive scaling factor:

$$a_{\text{adaptive}} \leftarrow a_{\text{min}} + (a_{\text{max}} - a_{\text{min}}) \cdot \sigma\big(\gamma(0.5 - \text{conf}_{\text{base}})\big)$$

▷ $\sigma(z) = 1/(1 + \exp(-z))$ is the sigmoid

Compute scaled logit for Novel class:

$$\ell_{\text{new}} \leftarrow \alpha_{\text{adaptive}} \cdot (\mathbf{x} \cdot \mathbf{z}_t^{(\text{new})})$$

Output: Scaled logit ℓ_{new} for the novel class

Table 10. Training epochs across datasets for different setups.

Setup	ImageNet	Caltech	Pets	Cars	Flowers	Food	Aircraft	SUN	DTD	EuroSAT	UCF
Few-shot	10	15	10	30	100	100	100	30	100	100	40
Base-to-Novel	15	12	10	30	100	100	100	30	100	60	40

cept for larger-scale ImageNet experiments, where we use a batch size of 32. All learnable parameters are initialized following standard practices. The text and image anchors, along with class-specific anchor weights, are initialized from a zero-mean Gaussian distribution with a standard deviation of 0.02. The shift parameters for both text and image branches are initialized to zero. Additionally, the adaptive scaling parameter a_{max} is fixed at 5 for all experiments. For Base-to-Novel and all-to-all few-shot settings, we report the training epochs for each dataset in Table 10. All reported results are averaged over three independent runs.

Adaptive Sharpness-Aware Minimization. To improve generalization for Novel classes, we employ ASAM, which extends Sharpness-Aware Minimization by introducing parameter-wise adaptive perturbations. Given model parameters $\mathbf{w}$ and loss $\mathcal{L}(\mathbf{w})$, ASAM seeks parameters that minimize the worst-case loss within a neighborhood:

$$\min_{\mathbf{w}} \max_{\|\boldsymbol{\epsilon}\| \le \rho} \mathcal{L}(\mathbf{w} + \boldsymbol{\epsilon}), \tag{18}$$

where ρ controls the neighborhood size.

Unlike SAM [15, 25], ASAM scales the perturbation adaptively based on the magnitudes of the parameters. For each parameter w_i, the perturbation is defined as:

$$\epsilon_i = (|w_i| + \eta) \cdot \frac{\rho}{\|\mathbf{g}\|_2} \cdot g_i, \tag{19}$$

where $\mathbf{g} = \nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w})$ is the gradient, $\|\mathbf{g}\|_2$ its ℓ_2 norm, and η is a small constant to avoid vanishing scaling.

The optimization proceeds in two steps:

- **Ascent step:** Perturb parameters along the adaptive direction:
$$\mathbf{w}^+ = \mathbf{w} + \boldsymbol{\epsilon}. \tag{20}$$
- **Descent step:** Revert the perturbation and update the parameters using the gradient evaluated at $\mathbf{w}^+$:
$$\mathbf{w} \leftarrow \mathbf{w} - \alpha \nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}^+), \tag{21}$$
where α is the learning rate.

Here, we compute the gradient norm as:

$$\|\mathbf{g}\|_2 = \left(\sum_i \|g_i\|_2^2 \right)^{1/2}, \tag{22}$$

ensuring a normalized perturbation direction across all parameters. The adaptive scaling $(|w_i| + \eta)$ biases the perturbation toward parameters with larger magnitudes, and promotes flatter minima in scale-invariant directions.

Role of Initialization in CASPA-A. CASPA-A employs an initialization strategy by constructing class prototypes through Gaussian-weighted aggregation [17] of prompt-based text embeddings. A shared set of prompt templates is used across all classes, where each template is instantiated with the corresponding class name. The resulting embeddings are normalized and combined using a Gaussian weighting scheme over the template set, which emphasizes central, more stable prompt formulations. Although the same templates are used for every class, diversity arises from the different semantic instantiations of the class names within each template. This reduces prompt-induced variance and improves the alignment between visual and textual representations.

Cross-data Transfer Mechanism. We train CASPA on all 1000 classes of the ImageNet dataset [5] using a 16-shot setting (ViT B/16). At inference time, for a target dataset whose distribution differs significantly from natural images, no additional training, fine-tuning, or access to target data

is performed, which ensures a strict zero-shot setting. Instead, text embeddings corresponding to the target classes are projected into the shared CLIP embedding space and subsequently adjusted using a learned anchor-induced bias along with a global shift term. The anchor vectors can be interpreted as a basis spanning semantically meaningful directions, while the learned class-conditioned weights determine how these directions are combined to transfer prior knowledge to unseen classes.

S1.3. Additional Results

CASPA demonstrates favorable plug-and-play compatibility with CoOp prompt- tuning, under the 16-shot Base-to-Novel generalization setting. As shown in Table 11, **CASPA + CoOp** outperforms $2SFS_{CoOp}$ in **24 out of 36** evaluation criteria (Base, Novel, HM), and surpasses CoOp in **33 out of 36** cases. Concretely, we integrate CoOp within CASPA by using the learned prompt embeddings to construct text features, which are then passed through the CASPA adaptation pipeline. In other words, CoOp operates at the input level by refining textual prototypes, while CASPA performs output-space adaptation via anchor banks, association weights, and residual shifts on top of the resulting image-text similarities. This modular composition allows CASPA to enhance CoOp without modifying its internal optimization or the frozen CLIP backbone.

We note that CASPA improves generalization performance, outperforming $2SFS_{CoOp}$ on **8 out of 12** Harmonic Mean (HM) evaluations, and CoOp on **11 out of 12** cases. Notable gains are observed on challenging datasets such as EuroSAT, DTD, UCF, and Food, where improvements in Novel class accuracy lead to a substantial increase in HM.

The radar chart in Figure 8 compares CASPA and CASPA-G performance across 11 datasets for Base, Novel, and HM accuracies. Overall, the two methods show largely comparable performance, with similar trends across most datasets. A notable exception is EuroSAT, where CASPA-G achieves a significantly higher HM; this is also discussed in the edge-case analysis in the main paper. This indicates its stronger generalization on this particular dataset, and CASPA-G can provide improved robustness on certain specialized domains.

Table 12 presents a comparison of CASPA with other state-of-the-art methods under the all-to-all generalization (16-shot) setting using ViT-L/14. CASPA-G achieves an overall performance of 87.1%, matching or surpassing the best performing methods across multiple datasets.

Ablation on Varying Source-Domain Training on Base-to-Novel Class Performance. The ablation study (Table 13) on ImageNet Base-to-Novel generalization using the CASPA + CoOp method with a ViT-B/16 backbone reveals a clear trade-off between Base and Novel class accuracies as the number of shots increases. Specifically, as

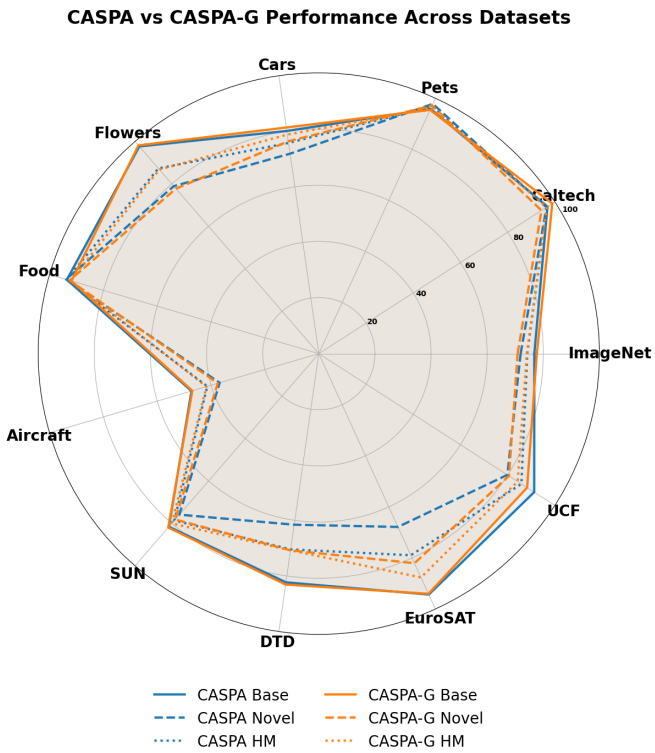

Figure 8. **Radar chart comparing CASPA and CASPA-G performance across 11 datasets.** Solid lines indicate Base accuracy, dashed lines indicate Novel class accuracy, and dotted lines indicate the Harmonic Mean (HM) of Base and Novel. CASPA is shown in blue, while CASPA-G is shown in orange. All values are in percentages.

the number of shots increases from 1 to 16, the Base accuracy consistently and significantly improves, rising from 70.20% at 1-shot to its peak of 76.44% at 16-shots. The most crucial insight from this study lies in the behavior of the Novel accuracy. Contrary to the typical risk in few-shot fine-tuning, where superior adaptation to the Base classes leads to overfitting and a subsequent degradation in performance on the truly unseen Novel classes, the Novel accuracy remains remarkably stable, fluctuating only narrowly between $\mathbf{65.29}\%$ and $\mathbf{65.57}\%$. This stability persists even as the Base accuracy surges by over 6 percent to $\mathbf{76.44}\%$ at 16-shots. This resilience strongly indicates that the design of CASPA has effectively reduced overfitting.

Qualitative Analysis on Most Confusing ImageNet Samples. In the main paper, we provide visualizations illustrating explainability [34] in Figures 2, 3, embedding analysis [38] in Figure 5, etc. Here, we further demonstrate CASPA's zero-shot predictive ability on challenging ImageNet Novel samples (Figure 7), where CLIP is prone to context-driven misclassification, but CASPA correctly identifies the objects. For example, in the first image, CLIP predicts 'desk' instead of the smaller desktop com-

Table 11. Comparison of plug-and-play capability of CASPA with CoOp-style prompt- tuning with CoOp and the latest SOTA method (2SFS + CoOp) on 16-shot Base-to-Novel generalization using ViT-B/16 across 11 datasets. Results are reported for Base, Novel, and Harmonic Mean (HM) accuracy. CASPA + CoOp exceeds the latest SOTA on **24/36** criteria, and exceeds CoOp on **33/36** criteria. CASPA outperforms 2SFS + CoOp on **8/12** and CoOp on **11/12** occasions on Harmonic Mean.

Dataset	Method		Accuracy		
			Base	Novel	HM
ImageNet	CoOp	*CVPR '22*	76.47	67.88	71.92
	$2SFS_{CoOp}$	*CVPR '25*	**77.44**	**71.11**	**74.14**
	$OURS_{CoOp}$		76.44	65.37	70.47
Caltech	CoOp	*CVPR '22*	98.00	89.81	93.73
	$2SFS_{CoOp}$	*CVPR '25*	98.00	**91.99**	**94.90**
	$OURS_{CoOp}$		**98.97**	90.79	94.70
Pets	CoOp	*CVPR '22*	93.67	95.29	94.47
	$2SFS_{CoOp}$	*CVPR '25*	93.35	**96.96**	95.12
	$OURS_{CoOp}$		**94.84**	96.13	**95.48**
Cars	CoOp	*CVPR '22*	78.12	60.40	68.13
	$2SFS_{CoOp}$	*CVPR '25*	**80.15**	67.87	73.50
	$OURS_{CoOp}$		79.01	**68.87**	**73.59**
Flowers	CoOp	*CVPR '22*	97.60	59.67	74.06
	$2SFS_{CoOp}$	*CVPR '25*	**98.16**	69.46	81.35
	$OURS_{CoOp}$		98.02	**76.06**	**85.65**
UCF	CoOp	*CVPR '22*	84.69	56.06	67.46
	$2SFS_{CoOp}$	*CVPR '25*	85.04	64.67	73.47
	$OURS_{CoOp}$		**87.65**	**75.86**	**81.33**
Food	CoOp	*CVPR '22*	88.33	82.26	85.19
	$2SFS_{CoOp}$	*CVPR '25*	88.06	88.68	88.37
	$OURS_{CoOp}$		**91.34**	**91.23**	**91.29**
Aircraft	CoOp	*CVPR '22*	40.44	22.30	28.75
	$2SFS_{CoOp}$	*CVPR '25*	44.60	**29.91**	**35.81**
	$OURS_{CoOp}$		**46.10**	29.27	35.80
SUN	CoOp	*CVPR '22*	80.60	65.89	72.51
	$2SFS_{CoOp}$	*CVPR '25*	79.16	**70.32**	**74.48**
	$OURS_{CoOp}$		**81.56**	66.36	73.18
DTD	CoOp	*CVPR '22*	79.44	41.18	54.24
	$2SFS_{CoOp}$	*CVPR '25*	81.40	49.11	61.27
	$OURS_{CoOp}$		**84.24**	**58.34**	**68.94**
EuroSAT	CoOp	*CVPR '22*	92.19	54.74	68.69
	$2SFS_{CoOp}$	*CVPR '25*	92.94	50.76	65.66
	$OURS_{CoOp}$		**93.56**	**76.42**	**84.13**
Average	CoOp	*CVPR '22*	82.69	63.22	71.66
	$2SFS_{CoOp}$	*CVPR '25*	83.49	68.27	75.12
	$OURS_{CoOp}$		**84.70**	**72.25**	**77.98**

puter, whereas, CASPA correctly predicts 'desktop computer' with 27.7% confidence. Similar improvements are seen for the envelope and espresso machine images. Please note that both predict in a zero-shot manner where neither are trained on those classes.

S1.4. Discussion

How does CASPA differ from prior adaptation methods? Unlike approaches that rely on additional modules such as MLPs [32, 33] or parameter-efficient tuning strategies like LoRA [12, 26], CASPA avoids introducing extra architectural components. Instead, it learns a compact set of shared semantic anchors that serve as reusable building blocks across classes.

Are the learned representations interpretable? Yes. The anchors correspond to meaningful semantic directions in joint embedding space, and provide interpretability into how class representations are constructed through combinations of these primitives (Figures 1 & 3 in the main paper).

What is the role of hyperparameters in CASPA's Novel transfer? Hyperparameters in CASPA are used to control the stability and sensitivity of different components of the model. Key hyperparameters, such as the number of anchors and the consistency weight, are selected via coarse grid search on ImageNet and then reused across datasets. This establishes a stable operating region that generalizes effectively to new tasks without requiring dataset-specific tuning.

In the case of confidence-adaptive scaling, the constant γ determines how sensitively the model adjusts its behavior based on Base class confidence. It controls how sharply the scaling factor responds to changes in confidence. A well-chosen value (e.g., $\gamma = 3$) ensures that the adjustment is neither too abrupt nor too weak.

How does CASPA achieve Base-to-Novel generalization, and can it be adjusted? CASPA generalizes from Base-to-Novel classes via shared semantic anchors, allowing knowledge transfer to unseen classes. Generalization can be tuned through anchor count, diversity regularization, and confidence-adaptive scaling. We also note that decreasing batch size increases Base accuracy (85.76%) while Novel accuracy decreases (76.81%), yielding a Harmonic Mean of 81.04%, showing the trade-off between Base specialization and Novel transfer.

Table 12. All-to-all generalization results (16-shots) using ViT-L/14.

Backbone	Method	ImageNet	SUN	Aircraft	ESAT	Cars	Food	Pets	Flowers	Caltech	DTD	UCF	Avg.
	Zero-Shot [31]	72.9	67.6	32.6	58.0	76.8	91.0	93.6	79.4	94.9	53.6	74.2	72.2
	CoOp [49]	78.2	77.5	55.2	88.3	89.0	89.8	94.6	99.1	97.2	74.4	87.3	84.6
	CoCoOp [48]	77.8	76.7	45.2	79.8	82.7	91.9	95.4	95.3	97.4	71.4	85.2	81.7
	TIP-Adapter-F [46]	79.3	79.6	55.8	86.1	88.1	91.6	94.6	98.3	97.5	74.0	87.4	84.8
	CLIP-Adapter [8]	76.4	78.0	46.4	75.8	83.8	91.6	94.3	97.3	97.3	71.3	86.1	81.7
	KgCoOp [42]	76.8	76.7	47.5	83.6	83.2	91.7	95.3	96.4	97.4	73.6	86.4	82.6
ViT-L/14	MaPLe [16]	78.4	78.8	46.3	85.4	83.6	92.0	95.4	97.4	97.2	72.7	86.5	83.1
	ProGrad [50]	78.4	78.3	55.6	89.3	88.8	90.8	94.9	98.7	97.5	73.7	87.7	84.9
	LP++ [13]	79.3	79.7	54.6	89.3	87.7	91.7	94.9	98.5	97.4	76.1	88.1	85.2
	MMA [41]	79.9	80.2	56.4	76.3	88.0	92.0	95.5	98.4	97.6	75.8	88.0	84.4
	2SFS [6]	79.4	**80.3**	**64.1**	**92.9**	**90.3**	91.1	95.5	99.1	97.5	**78.0**	89.5	**87.1**
	CASPA-G	**80.0**	79.8	62.7	92.0	89.7	**92.9**	**95.6**	**99.2**	**98.2**	77.7	**89.8**	**87.1**

Table 13. Ablation study on ImageNet Base-to-Novel generalization with CASPA + CoOp across different shots (1,2,4,8,16). Accuracy (%) is reported for Base, Novel, and Harmonic Mean (HM).

Shots	Accuracy (%)		
	Base	Novel	HM
1	70.20	65.46	67.75
2	72.24	**65.57**	68.74
4	73.42	65.52	69.24
8	75.14	65.29	69.87
16	**76.44**	65.37	**70.47**

Limitations and Future Directions. In this work, CASPA uses fixed hyperparameters across datasets, which limits dataset-specific optimization; tuning parameters such as anchor count, regularization strength, or confidence-adaptive scaling could further improve performance. The method also relies on learning meaningful and diverse anchors, and hence classes with highly unusual or complex semantics may require larger residual adjustments, potentially violating the low-rank assumption. Moreover, datasets with very few classes (e.g., EuroSAT) may underutilize the anchor bank, reducing compositional capacity and slightly limiting adaptation to Novel classes. Future work could explore adaptive hyperparameter tuning, dynamic anchor construction, and enhanced compositional mechanisms by amalgamating with methods like [2, 9, 10, 18, 23, 30, 44, 45] to further improve generalization and robustness across diverse datasets.

References

[1] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision (ECCV)*, pages 446–461. Springer, 2014. 5, 4

[2] Guangyi Chen, Weiran Yao, Xiangchen Song, Xinyue Li, Yongming Rao, and Kun Zhang. PLOT: Prompt learning with optimal transport for vision-language models. In *International Conference on Learning Representations (ICLR)*, 2023. 8

[3] Liang Chen, Ghazi Shazan Ahmad, Tianjun Yao, Lingqiao Liu, and Zhiqiang Shen. One last attention for your vision-language model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1464–1473, 2025. 2, 5, 6, 1

[4] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3606–3613. IEEE, 2014. 5, 4

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. IEEE, 2009. 5, 8, 1, 4

[6] Matteo Farina, Massimiliano Mancini, Giovanni Iacca, and Elisa Ricci. Rethinking few-shot adaptation of vision-language models in two stages. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 29989–29998, 2025. 2, 5, 6, 7, 8, 1

[7] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. In *2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW)*, pages 178–178. IEEE, 2004. 5, 4

[8] 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)*, 132(2):581–595, 2024. 2, 5, 7, 1, 8

[9] Yuncheng Guo and Xiaodong Gu. MMRL: Multi-modal representation learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 25015–25025, 2025. 2, 7, 8

[10] Yiwei Guo, Shaobin Zhuang, Kunchang Li, Yu Qiao, and Yali Wang. TransAgent: Transfer vision-language foundation models with heterogeneous agent collaboration. In *Advances in Neural Information Processing Systems 37*, 2024. 2, 7, 8

[11] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification.

IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 12(7):2217–2226, 2019. 5, 8, 4

[12] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022. 2, 7

[13] Yunshi Huang, Fereshteh Shakeri, Jose Dolz, Malik Boudiaf, Houda Bahig, and Ismail Ben Ayed. LP++: A surprisingly strong linear probe for few-shot clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23773–23782, 2024. 2, 5, 7, 1, 8

[14] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, pages 4904–4916. PMLR, 2021. 2, 1

[15] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations (ICLR)*, 2017. 5

[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 12345–12354, 2023. 1, 2, 5, 6, 7, 8

[17] Muhammad Uzair Khattak, Syed Talal Wasim, Muzammal 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, 8, 1, 5

[18] Muhammad Uzair Khattak, Muhammad Ferjad Naeem, Muzammal Naseer, Luc Van Gool, and Federico Tombari. Learning to prompt with text only supervision for vision-language models. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence (AAAI)*, pages 4230–4238, 2025. 2, 5, 8

[19] 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. 1

[20] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3D object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 554–561. IEEE, 2013. 5, 4

[21] Haoyang Li, Liang Wang, Chao Wang, Jing Jiang, Yan Peng, and Guodong Long. DPC: Dual-prompt collaboration for tuning vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 25623–25632, 2025. 2, 5, 6, 1

[22] Yilun Li, Miaomiao Cheng, Xu Han, and Wei Song. Divergence-enhanced knowledge-guided context optimization for visual-language prompt tuning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025. 2, 5, 6, 1

[23] Zheng Li, Xiang Li, Xinyi Fu, Xin Zhang, Weiqiang Wang, Shuo Chen, and Jian Yang. PromptKD: Unsupervised prompt distillation for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26617–26626. 2, 8

[24] Zheng Li, Yibing Song, Ming-Ming Cheng, Xiang Li, and Jian Yang. Advancing textual prompt learning with anchored attributes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3618–3627, 2025. 2, 5, 6, 8, 1

[25] Liangchen Liu, Nannan Wang, Xi Yang, Xinbo Gao, and Tongliang Liu. Surrogate prompt learning: Towards efficient and diverse prompt learning for vision-language models. In *Proceedings of the 42nd International Conference on Machine Learning*. PMLR, 2025. 2, 5

[26] Wenzhuo Liu, Fei Zhu, Longhui Wei, and Qi Tian. C-CLIP: Multimodal continual learning for vision-language model. In *Proceedings of the Thirteenth International Conference on Learning Representations (ICLR)*, 2025. 2, 7

[27] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 5, 4

[28] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP)*, pages 722–729. IEEE, 2008. 5, 4

[29] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3498–3505. IEEE, 2012. 5, 4

[30] Sarah Pratt, Ian Covert, Renjie Liu, and Ali Farhadi. What does a platypus look like? Generating customized prompts for zero-shot image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15691–15701, 2023. 2, 5, 8

[31] 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 38th International Conference on Machine Learning (ICML)*, pages 8748–8763, 2021. 1, 2, 3, 5, 6, 7, 4, 8

[32] Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958. 2, 7

[33] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986. 2, 7

[34] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via

gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017. 2, 6

[35] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. FLAVA: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16113–16123, 2022. 2, 1

[36] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 5, 4

[37] Yonglong Tian, Xinlei Chen, Surya Ganguli, Phillip Isola, and Dilip Krishnan. What makes for good views for contrastive learning? In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6827–6839, 2020. 1

[38] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9 (86):2579–2605, 2008. 6, 7

[39] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3485–3492, 2010. 5, 4

[40] Chen Xu, Yuhan Zhu, Haocheng Shen, Boheng Chen, Yixuan Liao, Xiaoxin Chen, and Limin Wang. Progressive visual prompt learning with contrastive feature re-formation. *International Journal of Computer Vision (IJCV)*, pages 511–526, 2025. 2, 5, 6, 1

[41] Lingxiao Yang, Ru-Yuan Zhang, Yanchen Wang, and Xiaohua Xie. MMA: Multi-modal adapter for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23826–23837, 2024. 2, 5, 6, 7, 8

[42] 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, 5, 6, 7, 8

[43] Hantao Yao, Rui Zhang, and Changsheng Xu. TCP: Textual-based class-aware prompt tuning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23438–23448, 2024. 2, 5, 1

[44] Tao Yu, Zhihe Lu, Xin Jin, Zhibo Chen, and Xinchao Wang. Task residual for tuning vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10899–10909, 2023. 2, 8

[45] Ji Zhang, Shihan Wu, Lianli Gao, Heng Tao Shen, and Jingkuan Song. DePT: Decoupled prompt tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12924–12933, 2024. 2, 8

[46] 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 *European Conference on Computer Vision (ECCV)*, pages 493–510, 2022. 2, 5, 7, 1, 4, 8

[47] Cairong Zhao, Yubin Wang, Xinyang Jiang, Yifei Shen, Kaitao Song, Dongsheng Li, and Duoqian Miao. Learning domain invariant prompt for vision-language models. *IEEE Transactions on Image Processing (TIP)*, pages 1348–1360, 2024. 2, 6

[48] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16816–16825, 2022. 1, 2, 6, 7, 8

[49] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision (IJCV)*, pages 2337–2348, 2022. 2, 5, 6, 7, 8, 1, 4

[50] 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, 5, 6, 7, 1, 8