TEXT-CONDITIONED ORTHOGONAL COMPENSATION FOR CLIP-BASED FEW-SHOT ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

CLIP exhibits strong transferability in few-shot adaptation, yet two key challenges remain. First, existing adaptation methods are structurally diverse, while a unified geometric understanding of their underlying connections is still lacking. Second, effective adaptation should capture task-specific information while preserving the inherent category flexibility of CLIP, which is often weakened by label-space-specific adaptation. To address these challenges, we introduce a modality-gap-inspired geometric framework that interprets major CLIP adaptation strategies through the visual–classifier mismatch, and our subspace analysis further motivates complementary classifier directions beyond the principal text-classifier subspace. Based on this insight, we propose Text-Conditioned Orthogonal Compensation (TCOC), which uses a shared text-conditioned generator to produce class-wise residuals and projects them onto the orthogonal complement of the principal text-classifier subspace. This design retains the original text classifier as a semantic anchor while introducing complementary task-specific directions. Since the generator is shared across classes, TCOC can naturally extend to new label spaces. Experiments on standard few-shot classification and multiple generalization settings demonstrate its effectiveness and category flexibility. The code will be publicly released upon acceptance.

1 INTRODUCTION

Vision-Language Models, such as CLIP (Radford et al., 2021), have demonstrated strong zero-shot generalization by aligning visual and textual representations (Yang et al., 2023; Zhou et al., 2025). Unlike conventional vision models with fixed classification heads, CLIP dynamically constructs classifiers from class names through the text encoder. This enables *category-flexible* inference, allowing CLIP to accommodate label spaces with different class numbers and semantics without introducing new class-indexed parameters. Nevertheless, the original zero-shot text classifier often adapts insufficiently to downstream tasks under limited supervision, motivating CLIP-based few-shot adaptation (Zhu et al., 2023b). An effective adaptation method should exploit task-relevant visual information from a few labeled samples while preserving this intrinsic category flexibility.

A broad range of CLIP-based few-shot adaptation methods have been developed, including prompt tuning (Zhou et al., 2022b;a), feature adapters (Gao et al., 2024), support-set caches (Zhang et al., 2022), and classifier adaptation (Yu et al., 2023). Despite their strong performance, two fundamental questions remain. First, these structurally different approaches modify prompts, features, caches, or classifiers in distinct ways, raising the question of **whether these methods can be understood under a unified geometric perspective.** Second, **can the resulting geometric understanding be translated into an adaptation mechanism that captures task-specific information while preserving the category flexibility?** This challenge is particularly relevant to many recent competitive methods introducing label-space-specific caches or classifier (Tang et al., 2024; Bendou et al., 2025). Although effective in closed-set adaptation, these modules are structurally tied to the training label space and cannot be directly reused when the class identities or class numbers change (Fig. 1(a)).

To develop such a geometric understanding, we revisit few-shot adaptation through the lens of the modality gap. Prior studies have shown that, although CLIP maps images and texts into a shared embedding space, representations from the two modalities often occupy separated regions (Liang et al., 2022) (Fig. 1(b)). When text features are directly used as classifiers, their spanned subspace

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Figure 1: (a) Label-space-specific caches and classifiers bind adaptation to fixed class sets, limiting generalization to unseen classes. (b) CLIP exhibits an inherent modality gap.

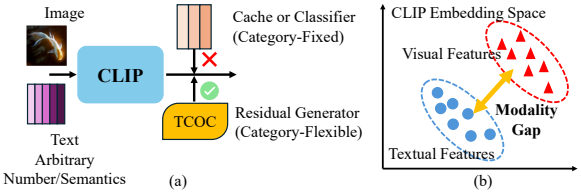


Table 1: Post-hoc subspace-gap analysis of CLIP few-shot adaptation methods on Flowers102 under the 16-shot setting.

Method	Type	$\mathcal{G}_{\text{realized}} \downarrow$	$\mathcal{G}_{\text{induced}} \downarrow$
CLIP	–	0.7813	–
CoOp	Prompt	0.7395	–
CLIP-Adapter	Adapter	0.7532	–
TaskRes	Classifier	0.7548	0.6192
Tip-Adapter	Cache	–	0.6457
ProKeR	Cache	–	0.6953
TCOC (Ours)	Classifier	**0.7183**	**0.5877**

may not fully cover the task-relevant directions present in downstream visual features. We refer to this insufficient coverage as the ***visual–classifier mismatch***, viewing it as a task-level manifestation of the modality gap. Under this geometric view, prompt-based methods modify the text classifier, adapter-based methods reshape the feature space, while cache-based and classifier-based methods introduce additional decision directions. This provides a unified geometric interpretation of these structurally different strategies as different ways of mitigating the visual–classifier mismatch.

Our subspace analysis further suggests that the principal text-classifier subspace may leave task-relevant visual directions underrepresented, motivating classifier adaptation along complementary directions beyond this subspace. However, directly learning such directions as a free residual matrix inevitably ties the adapted parameters to the identities and number of the training classes. This reveals both a key insight and a fundamental tension: effective adaptation should introduce complementary task-specific directions while parameterizing them independently of class indices.

Based on these, we propose Text-Conditioned Orthogonal Compensation (TCOC), a category-flexible classifier adaptation method. Instead of learning a class-indexed residual, TCOC projects class text features into a common coordinate space and applies a shared text-conditioned generator to produce class-wise residuals, making the learned adaptation independent of class identities. The generated residuals are further projected onto the orthogonal complement of the principal text-classifier subspace, introducing complementary task-specific directions while retaining the original text classifier as a semantic anchor and thereby mitigating the visual–classifier mismatch. When the label space changes, the learned generator can be directly reused. Our contributions are threefold:

- We present a modality-gap-inspired geometric framework that unifies major CLIP adaptation strategies through the visual–classifier mismatch and reveals complementary discriminative directions beyond the principal text-classifier subspace.
- Guided by this insight, we propose TCOC, which combines shared text-conditioned residual generation with orthogonal compensation for category-flexible adaptation.
- Experiments across few-shot classification and multiple generalization settings demonstrate strong performance and category flexibility of TCOC.

2 Related Work

CLIP-based Few-Shot Adaptation. Existing methods can be broadly categorized into prompt-based, adapter-based, cache-based, and classifier-based approaches. Prompt-based methods, such as CoOp (Zhou et al., 2022b) and CoCoOp (Zhou et al., 2022a), optimize textual contexts to adapt the text classifier. Adapter-based methods introduce modules to adjust downstream representations, such as CLIP-Adapter (Gao et al., 2024). Cache-based methods exploit support features and labels through task-specific caches, including Tip-Adapter (Zhang et al., 2022) and its kernel-regression extension ProKeR (Bendou et al., 2025). Classifier-based methods instead learn additional class-wise residuals or classifiers (Yu et al., 2023; Tang et al., 2024; Wang et al., 2024). Although effective in closed-set adaptation, many cache- and classifier-based methods instantiate label-specific components, preventing their reuse when the class identities change. Parameter-efficient fine-tuning (PEFT) methods, such as CLIP-LoRA (Zanella & Ben Ayed, 2024) and CLIP-SVD (Koleilat et al., 2026), adapt a subset of pretrained parameters. In contrast, TCOC parameterizes classifier-side correction through a shared text-conditioned function rather than class-indexed components, preserving category flexibility with frozen encoders while remaining complementary to backbone PEFT.

Modality Gap. Prior work (Liang et al., 2022) shows that, although CLIP maps images and texts into a shared embedding space, features from different modalities often reside in two separated narrow cones, forming an inherent modality gap. Subsequent works reveal that this gap can affect downstream tasks such as zero-shot recognition, cross-modal retrieval, domain adaptation, and attempt to regulate it through alignment or compensation strategies (Eslami & de Melo, 2025; Mistretta et al., 2025; Schrodi et al., 2025; Cai et al., 2025; Yamaguchi et al., 2025). We develop a geometric framework for few-shot adaptation from the modality-gap perspective, characterizing the gap's task-level effect as a visual–classifier mismatch. While MG-CLIP (Huang et al., 2025) compensates for the gap in continual learning using an auxiliary visual classifier, TCOC explicitly restricts text-conditioned classifier residuals to the orthogonal complement of the principal text-classifier subspace.

3 Revisiting CLIP-based Few-Shot Adaptation

3.1 Preliminaries

We focus on CLIP-based few-shot adaptation throughout the following analysis. CLIP consists of an image encoder $f(\cdot)$ and a text encoder $g(\cdot)$, which map images and texts into a D-dimensional embedding space. For a downstream task with training label space $\mathcal{Y} = \{1, \dots, C\}$, the K-shot support set is denoted by $\mathcal{D}_s = \{(I_i, y_i)\}_{i=1}^{N}, N = CK$, where I_i is an image and $y_i \in \mathcal{Y}$ is its label. For each class $c \in \mathcal{Y}$, its name is inserted into a text template $p(\cdot)$ to obtain the normalized text feature $t_c = \frac{g(p(c))}{\|g(p(c))\|_2} \in \mathbb{R}^D$. Subsequently, stacking all class features yields the text classifier $T = [t_1, \dots, t_C]^\top \in \mathbb{R}^{C \times D}$. Therefore, given I_i, its normalized visual feature is $x_i = \frac{f(I_i)}{\|f(I_i)\|_2} \in \mathbb{R}^D$, and the corresponding CLIP logits are $z_i = \tau x_i T^\top \in \mathbb{R}^C$, where τ denotes the logit scale. Few-shot adaptation uses $\mathcal{D}_s$ to improve downstream prediction over the original text classifier (Li et al., 2025a). We next examine this adaptation process through a geometric perspective.

3.2 A Modality-Gap-Inspired Geometric Perspective

Although CLIP maps images and texts into a shared feature space, their representations may remain systematically separated, forming a modality gap. When text features are used as classifier weights, the subspace spanned by the text classifier may not fully cover the discriminative directions of downstream visual features. We refer to this subspace-level discrepancy as the visual–classifier mismatch. To characterize this mismatch geometrically, we first extract a basis for the principal text-classifier subspace and compare its expressive capacity with an ideal downstream classifier. Specifically, we perform singular value decomposition (SVD) on the text classifier $T = U_T \Sigma_T V_T^\top$. Let $B_P \in \mathbb{R}^{D \times r_p}$ contain the top-r right singular vectors of T. We refer to $\mathrm{span}(B_P)$ as the principal text-classifier subspace, with orthogonal projector $P_T = B_P B_P^\top$. We then introduce a hypothetical oracle linear classifier $W^\star \in \mathbb{R}^{C \times D}$ solely to characterize the expressive limitation of this subspace. This leads to the following result.

Proposition 1. *Let $\mathcal{W}_T = \left\{ W_T = L B_P^\top \;\middle|\; L \in \mathbb{R}^{C \times r_p} \right\}$ be the set of classifiers whose rows lie in $\mathrm{span}(B_P)$. The best approximation to $W^\star$ within $\mathcal{W}_T$ is $\widehat{W}_T = W^\star P_T$, with approximation error:*

$$\mathcal{E}_T = \min_{W_T \in \mathcal{W}_T} \|W_T - W^\star\|_F^2 = \|W^\star (I - P_T)\|_F^2. \tag{1}$$

The proof is provided in the **Appendix A.1**. Proposition 1 shows that restricting the classifier to the principal text-classifier subspace leaves the oracle component $W^\star(I - P_T)$ unrepresented. Therefore, the orthogonal complement of this subspace provides additional capacity for classifier components associated with downstream visual discrimination.

From this geometric perspective, most adaptation methods can be interpreted as mitigating the visual–classifier mismatch through different components of the decision process:

- Prompt-based methods move text-classifier directions by optimizing textual contexts.
- Adapter-based methods reshape visual or textual features, altering their geometry with classifiers.
- Cache-based methods incorporate support visual features as additional decision directions.
- Classifier-based methods introduce task-specific classifier directions to correct the text classifier.

To provide empirical evidence for the above geometric perspective, we introduce a post-hoc directional subspace gap that measures how well a method's decision space covers the principal directions of visual features. Given a visual feature matrix X and a decision-direction matrix A, whose rows represent final classifier directions or additional induced directions, we first normalize their rows and perform SVD. We retain the principal subspaces that preserve 95% of the cumulative energy, with orthonormal bases $B_X = [b_1^X, \ldots, b_{r_X}^X] \in \mathbb{R}^{D \times r_X}$ and $B_A \in \mathbb{R}^{D \times r_A}$. The directional gap from the visual subspace to the decision subspace is defined as (Huang et al., 2025):

$$\mathcal{G}(X, A) = \frac{1}{r_X} \sum_{j=1}^{r_X} \left\| b_j^X - B_A B_A^\top b_j^X \right\|_2 . \tag{2}$$

This asymmetric measure evaluates how well the decision subspace covers the visual feature directions. A smaller value indicates a weaker directional mismatch. Since adaptation methods modify different components of the decision process, we consider two types of gap. For methods whose predictions are represented by visual features X_m and an explicit classifier W_m, the realized gap is:

$$\mathcal{G}_{\text{realized}} = \mathcal{G}(X_m, W_m). \tag{3}$$

For the original CLIP, $X_m = X$ and $W_m = T$. Prompt-based methods produce an adapted classifier W_m, whereas visual CLIP-Adapter uses adapted features X_m. Classifier-based methods are evaluated using their final combined classifier W_m. Thus, the realized gap characterizes the geometry actually used for prediction. For methods that introduce additional directions through caches, residuals, or auxiliary classifiers, we denote these directions by Q_m and define the induced gap as:

$$\mathcal{G}_{\text{induced}} = \mathcal{G}\big(X_m, [T; Q_m]\big), \tag{4}$$

where $[T; Q_m]$ denotes the joint subspace spanned by the original text classifier and the induced directions. For cache-based methods, Q_m represents the additional decision space introduced by the cache; for classifier-based methods, it represents the learned classifier or residual directions.

Table 1 presents a post-hoc analysis on Flowers102 under the 16-shot setting. Relative to the original CLIP, representative adaptation methods reduce their corresponding realized or induced gaps. These observations support a geometric view of few-shot adaptation through changes in the visual–classifier mismatch. Since the two gaps describe different decision forms and their retained subspace ranks may differ, the results are intended to characterize geometric changes rather than provide a unified ranking across columns. This feature-space perspective is complementary to the logit-bias view of AMU-Tuning (Tang et al., 2024), which describes adaptation as decision-level corrections to the original CLIP logits. Our analysis instead provides a geometric interpretation of such corrections through the relationship between visual representations and classifier directions.

3.3 Category Flexibility in Few-Shot Adaptation

Besides mitigating the visual–classifier mismatch, few-shot adaptation should retain the category flexibility of CLIP. Given a new label space $\mathcal{Y}'$, CLIP can directly re-encode its class names to construct a new text classifier $T' \in \mathbb{R}^{C' \times D}$. However, many cache-based and classifier-based methods introduce label-space-specific components. For example, caches depend on the current label assignments, while class-wise residuals or classifiers $\Delta W \in \mathbb{R}^{C \times D}$ are indexed by training-class identities. Such components cannot be directly reused when class identities or class numbers change. This motivates a shared text-conditioned adaptation function that can generate classifier corrections for arbitrary label spaces without relying on class-indexed parameters.

4 Text-Conditioned Orthogonal Compensation

Motivated by the modality-gap-inspired geometric analysis and the need for category-flexible adaptation, we propose TCOC, which reformulates classifier adaptation through shared text-conditioned residual generation with geometric orthogonality constraints. Given a text classifier T constructed from an arbitrary label space, TCOC maps each class text feature into a shared text-coordinate space and applies the same generator row-wise to produce class-wise residuals. The generated residuals are then projected onto the orthogonal complement of the principal text-classifier subspace, yielding $R_\perp$. The adapted classifier is $W = \mathrm{Norm}(T + \beta R_\perp)$, where $\mathrm{Norm}(\cdot)$ denotes row-wise ℓ_2 normalization. TCOC therefore retains the original text classifier as a semantic anchor while introducing complementary classifier directions for few-shot adaptation. The pipeline is shown in Fig. 2.

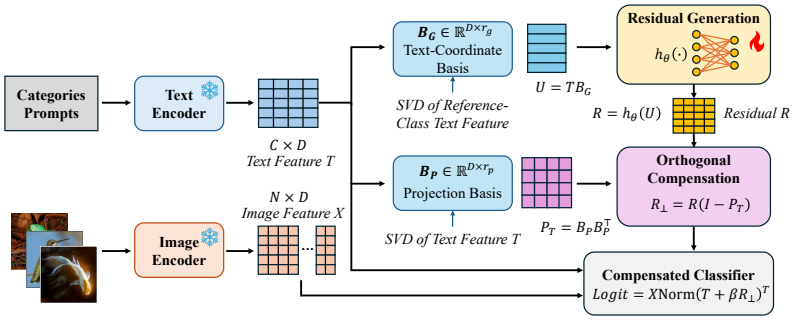


Figure 2: Overview of TCOC. A text-conditioned generator produces residuals from class text coordinates for category-flexible adaptation. The orthogonal compensation module then projects these residuals onto the orthogonal complement of the principal text-classifier subspace, mitigating the visual–classifier mismatch while retaining the original text classifier as a semantic anchor.

4.1 Text-Conditioned Residual Generation

Instead of directly optimizing a free class-indexed residual matrix, TCOC generates each class-wise residual through a shared function conditioned on its text representation. Let $T_{\text{ref}} \in \mathbb{R}^{C_{\text{ref}}\times D}$ denote the text classifier constructed from a reference label set. We perform SVD on T_{ref} and retain its leading r_g right singular vectors to construct the text-coordinate basis $B_G \in \mathbb{R}^{D\times r_g}$, where $r_g = \min\{C_{\text{ref}}, D, d_g\}$, and d_g denotes a predefined maximum coordinate dimension. Given the text classifier $T \in \mathbb{R}^{C\times D}$ for the current few-shot task label space, its class features are represented in the reference coordinate system as:

$$U = TB_G \in \mathbb{R}^{C\times r_g}. \tag{5}$$

The c-th row u_c encodes the coordinates of class c along the principal reference text-classifier directions. We then apply a shared residual generator $h_\theta(\cdot)$ independently to each row. Stacking the resulting class-wise residuals gives:

$$R = h_\theta(U) = [h_\theta(u_1), h_\theta(u_2), \ldots, h_\theta(u_C)]^\top \in \mathbb{R}^{C\times D}. \tag{6}$$

In implementation, h_θ combines a linear branch and a lightweight MLP branch to map text coordinates to classifier residuals. Since the generator is applied row-wise and contains no parameters indexed by class identities, it can accommodate label spaces with different numbers and semantics of classes. By default, we construct T_{ref} from the few-shot training classes. Thus, $T_{\text{ref}} = T$ in standard few-shot adaptation, whereas during transfer B_G and h_θ remain fixed and T' is reconstructed from the new class names. We investigate external reference label sets in the **Appendix D.2**.

4.2 Orthogonal Compensation

The residual R in Eq. 6 provides a class-wise classifier correction. However, directly adding R to the original text classifier may introduce substantial overlap with its dominant directions, reducing the complementarity of the learned correction. We therefore constrain the residual to the orthogonal complement of the principal text-classifier subspace associated with the current label space.

Let T denote the text classifier constructed from the training classes. We perform SVD on T and retain its leading r_p right singular vectors to form the orthonormal projection basis $B_P \in \mathbb{R}^{D\times r_p}$. The corresponding projector onto the principal training-class text-classifier subspace is $P_T = B_P B_P^\top$. When generalizing to a new label space, the new text classifier T' is reconstructed from its class names, and B'_P is recomputed accordingly to maintain the orthogonality of the compensation. The following corollary extends Proposition 1 by considering an additional residual subspace.

Corollary 1. *Let $\mathcal{S}_T = \text{span}(B_P)$, and let P_R be the orthogonal projector onto a residual subspace $\mathcal{S}_R \subseteq \mathcal{S}_T^\perp$. If the rows of the classifier are constrained to $\mathcal{S}_T \oplus \mathcal{S}_R$, the best approximation to the oracle classifier $W^\star$ is $W^\star_\oplus = W^\star(P_T + P_R)$, with approximation error:*

$$\mathcal{E}_\oplus = \|W^\star(I - P_T - P_R)\|_F^2. \tag{7}$$

Compared with the text-subspace approximation error $\mathcal{E}_T$ in Proposition 1, we have $\mathcal{E}_\oplus \le \mathcal{E}_T$, and:

$$\mathcal{E}_T - \mathcal{E}_\oplus = \|W^\star(I - P_T)P_R\|_F^2. \tag{8}$$

The proof is provided in the **Appendix A.2**. This result shows that introducing an orthogonal residual subspace enlarges the feasible classifier space and cannot increase the best linear approximation error. The improvement is nonzero when the residual subspace captures useful components of $W^\star$ outside the principal text-classifier subspace. Motivated by this, we project the generated residual matrix onto the orthogonal complement of the principal text-classifier subspace:

$$R_\perp = R(I - P_T) = R - (RB_P)B_P^\top. \tag{9}$$

By construction, $R_\perp B_P = 0$, so the learned compensation contains no component along the protected principal directions. The adapted classifier W is then constructed as

$$\widetilde{W} = T + \beta R_\perp, \qquad W = \mathrm{Norm}(\widetilde{W}), \tag{10}$$

where β controls the compensation strength. Therefore, TCOC actively learns compensation directions that capture task-relevant visual discriminative information, thereby mitigating the visual–classifier mismatch while retaining the original text classifier as a semantic anchor.

4.3 Training and Inference

During training, we freeze both the CLIP image and text encoders and optimize only the text-conditioned residual generator $h_\theta(\cdot)$. Given the few-shot support set $\mathcal{D}_s = \{(I_i, y_i)\}_{i=1}^N$, we extract the normalized image features x_i and construct the text classifier T from the training classes. TCOC then generates the orthogonal residual $R_\perp$ and constructs the normalized adapted classifier W. The generator is optimized using the standard cross-entropy loss:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{\exp(\tau x_i w_{y_i}^\top)}{\sum_{c=1}^{C} \exp(\tau x_i w_c^\top)}, \tag{11}$$

where w_c denotes the normalized classifier weight of class c in W, and τ is the logit scale. For standard few-shot inference, the adapted classifier is directly evaluated on the training label space. When the test label space changes to $\mathcal{Y}'$, we re-encode the new class names to construct $T' \in \mathbb{R}^{C' \times D}$. We further perform SVD on T' and retain its leading r_p right singular vectors to construct the projection basis $B'_P \in \mathbb{R}^{D \times r_p}$. The learned compensation mechanism is then applied as:

$$W' = \mathrm{Norm}(T' + \beta\, h_\theta(T'B_G)(I - B'_P {B'_P}^\top)). \tag{12}$$

Here, B_G and h_θ remain fixed, while T' and its projection basis B'_P are constructed from the new class names. This guarantees that the generated compensation is orthogonal to the principal text directions of the inference label space. The pseudocode is summarized in **Appendices B.2 and B.3**.

5 Experiments

5.1 Experimental Setup

Datasets. We evaluate TCOC on 11 standard image classification benchmarks, including ImageNet (Deng et al., 2009), Caltech101 (Fei-Fei et al., 2004), OxfordPets (Parkhi et al., 2012), StanfordCars (Krause et al., 2013), Flowers102 (Nilsback & Zisserman, 2008), Food101 (Bossard et al., 2014), FGVCAircraft (Maji et al., 2013), SUN397 (Xiao et al., 2010), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019), and UCF101 (Soomro et al., 2012). We further use four ImageNet variants, i.e., ImageNet-V2 (Recht et al., 2019), ImageNet-Sketch (Wang et al., 2019), ImageNet-A (Hendrycks et al., 2021b), and ImageNet-R (Hendrycks et al., 2021a), to evaluate robustness under domain shifts. Following common practice in CLIP few-shot adaptation, we conduct three random runs and report the average accuracy over the 11 benchmarks under different shot settings.

Evaluation protocol. We evaluate TCOC under both few-shot classification and generalization settings. For few-shot classification, we compare TCOC with category-flexible adaptation methods that keep the encoders frozen, including prompt-based methods such as CoOp, CoCoOp, KgCoOp (Yao et al., 2023), ProGrad (Zhu et al., 2023a), MaPLe (Khattak et al., 2023), TCP (Yao et al., 2024), and DeKgCoOp (Li et al., 2025b), as well as the adapter-based CLIP-Adapter. We then compare TCOC with label-space-specific cache- and classifier-based methods, including LP++ (Huang et al., 2024),

Table 2: Few-shot comparison with category-flexible methods on 11 datasets.

Method	Shots					Avg.
	1	2	4	8	16	
Zero-shot CLIP	65.36	65.36	65.36	65.36	65.36	65.36
CoOp	68.09	70.13	73.59	76.45	79.01	73.45
CoCoOp	66.95	67.63	71.98	72.92	75.02	70.90
CLIP-Adapter	67.87	70.20	72.65	76.92	79.86	73.50
KgCoOp	69.51	71.57	74.48	75.82	77.26	73.73
ProGrad	68.20	71.78	74.21	77.93	79.20	74.26
MaPLe	69.27	72.58	75.37	78.89	81.79	75.58
DeKgCoOp	69.77	73.02	75.12	77.94	80.81	75.33
TCP	70.63	73.59	76.07	78.39	80.98	75.93
TCOC (Ours)	**72.00**	**74.34**	**76.85**	**79.87**	**82.09**	**77.03**

Table 3: Few-shot comparison with label-space-specific adaptation and PEFT methods.

Method	Shots					Avg.
	1	2	4	8	16	
LP++	70.35	72.93	75.77	77.94	80.32	75.46
Tip-Adapter	68.89	70.42	72.69	74.41	76.44	72.57
Tip-Adapter-F	70.62	73.08	75.75	78.51	81.15	75.82
GDA	69.39	73.09	76.24	79.71	81.70	76.03
ProKeR	71.32	73.74	76.23	79.84	82.01	76.63
AdaLoRA	69.04	72.21	75.50	78.13	80.95	75.17
CLIP-LoRA	72.16	75.29	77.28	80.02	82.83	77.52
CLIP-SVD	73.20	76.06	78.18	80.55	82.97	78.19
TCOC (Ours)	72.00	74.34	76.85	79.87	82.09	77.03
+CLIP-SVD	**74.02**	**76.98**	**78.94**	**81.37**	**83.71**	**79.00**

Table 4: Domain generalization accuracy (%). Methods are trained on ImageNet1K with 16 shots and evaluated on ImageNet variants. Avg. denotes the mean accuracy over the source and targets.

Method	Source	Target				**Avg.**
	IN	IN-V2	IN-S	IN-A	IN-R	
CLIP	66.72	60.83	46.15	47.77	73.96	59.09
CoCoOp	71.02	64.07	48.75	50.63	76.18	62.13
KgCoOp	71.20	64.10	48.97	50.69	76.70	62.33
ProGrad	72.24	64.73	47.61	49.39	74.58	61.71
MaPLe	70.72	64.07	49.15	50.90	76.98	62.36
Tip-Adapter	53.81	45.69	29.21	36.04	55.26	44.00
TaskRes	70.84	62.15	43.76	43.91	71.59	58.45
ProKeR	72.52	64.82	49.03	50.12	76.75	62.65
TCOC (Ours)	**73.12**	**65.15**	**49.23**	**50.94**	**77.50**	**63.19**

Tip-Adapter, GDA (Wang et al., 2024), and ProKeR. We further include PEFT methods such as AdaLoRA (Zhang et al., 2023), CLIP-LoRA, and CLIP-SVD. Since TCOC introduces compensation on the classifier side, it can be combined with PEFT methods to examine their complementarity.

We evaluate generalization from three perspectives: domain generalization, base-to-novel generalization, and cross-dataset transfer. For domain generalization, methods are trained on ImageNet1K under the 16-shot setting and directly evaluated on ImageNet variants that share the same label space. Therefore, both category-flexible and label-space-specific methods can be compared. For base-to-novel generalization, methods are trained on base classes and evaluated on both base and novel classes. For cross-dataset transfer, methods are trained on ImageNet1K and directly transferred to the other ten datasets. Since the test label space changes in the latter two settings, we mainly compare category-flexible adaptation methods with frozen encoders.

Implementation details. For fair comparison, all main experiments adopt ViT-B/16 CLIP as the backbone and follow the data splits and evaluation protocol of CoOp. Hyperparameters are selected on the corresponding validation sets, and final performance is reported on the full test sets. For baseline methods, we use published results in CLIP-SVD when the experimental settings are consistent with ours; otherwise, we reproduce them using their official implementations. In TCOC, the residual generator consists of a parallel linear branch and a lightweight MLP branch, with the MLP hidden dimension set to 128. The model is trained for 50 or 100 epochs using the Adam optimizer with a batch size of 256. The initial learning rate is selected from $\{10^{-3}, 5 \times 10^{-4}, 10^{-4}\}$. We use $\beta = 1.0$ for all training settings. For the text-coordinate basis, we set the maximum coordinate dimension to $d_g = 512$, with $r_g = \min\{C_{\text{ref}}, D, d_g\}$. The projection rank is set to $r_p = 4$ for all datasets except ImageNet1K, for which $r_p = 8$ is used. More details are provided in **Appendix C**.

5.2 Few-shot Comparison

Table 2 first compares TCOC with category-flexible adaptation methods that keep the CLIP encoders frozen. TCOC achieves the best performance under all shot settings, reaching an average accuracy of 77.03% across the five settings and outperforming the strongest baseline by 1.10%. These indicate that the shared text-conditioned generator can exploit limited supervision to produce effective

Table 5: Cross-dataset transfer accuracy (%). Methods are trained on ImageNet1K with 16 shots and evaluated on the other ten datasets. Avg. denotes the mean accuracy over the source and targets.

	Source	**Avg.**	Target									
Method	IN		Caltech	Pets	Cars	Flowers	Food	Aircraft	SUN	DTD	EuroSAT	UCF
CLIP	66.72	65.30	92.98	89.13	65.29	71.30	86.11	24.90	62.59	44.56	47.84	66.83
CoCoOp	71.02	66.22	**94.43**	90.14	65.32	71.88	86.06	22.94	**67.36**	45.73	45.37	68.21
KgCoOp	70.66	65.98	93.92	89.83	65.41	70.01	86.36	22.51	66.16	46.35	46.04	68.50
ProGrad	72.24	63.57	91.52	89.64	62.39	67.87	85.40	20.61	62.47	39.42	43.46	64.29
MaPLe	70.72	66.70	93.53	90.49	65.57	**72.23**	86.20	24.74	67.01	**46.49**	48.06	68.69
DeKgCoOp	71.34	65.93	93.87	90.16	65.91	70.60	86.37	23.37	66.11	46.16	43.15	68.17
TCP	71.40	66.76	93.97	**91.25**	64.69	71.21	86.69	23.45	67.15	44.35	51.45	68.73
TCOC (Ours)	**73.12**	**67.29**	94.05	89.91	**65.98**	71.25	**86.81**	**25.55**	67.23	45.21	**52.30**	**68.80**

Table 6: Base-to-novel generalization accuracy (%). Methods are trained on base classes and evaluated on base and novel classes. HM denotes the harmonic mean of two accuracies.

Acc.	CLIP	KgCoOp	ProGrad	MaPLe	DeKgCoOp	TCP	TCOC
Base	69.34	80.73	82.48	82.28	82.59	84.13	**84.64**
Novel	74.22	73.60	70.75	75.14	74.93	75.36	**76.36**
HM	71.70	77.00	76.16	78.55	78.57	79.51	**80.29**

class-wise residuals, while orthogonal compensation further introduces complementary discriminative directions, leading to consistently strong few-shot performance without sacrificing category flexibility. Table 3 further compares TCOC with label-space-specific adaptation and PEFT methods. Despite avoiding class-specific caches and classifiers, TCOC outperforms strong methods such as ProKeR and GDA. When used as a classifier-side plug-in for CLIP-SVD, TCOC further improves the average accuracy to 79.00% and achieves the best result. This demonstrates the complementarity between classifier-side orthogonal compensation and parameter-efficient backbone adaptation.

5.3 Generalization Evaluation

We further evaluate TCOC from three perspectives: domain generalization, base-to-novel generalization, and cross-dataset transfer. For domain generalization and cross-dataset transfer, we include both source- and target-domain performance in the average metric, so as to jointly assess source-task adaptation and transfer robustness. Table 4 reports the domain generalization results from ImageNet1K to its variants. TCOC achieves strong performance on both the source domain and the target domains, obtaining the best average accuracy of 63.19%. It outperforms ProKeR and KgCoOp by 0.54% and 0.86%. These results show that TCOC learns compensation that remains robust under domain shifts while preserving strong source-domain adaptation.

Tables 5, 6 further evaluate generalization under changes in the label space. In this setting, label-space-specific methods are no longer applicable. In cross-dataset transfer, TCOC is trained on ImageNet1K and directly transferred to the other ten datasets, achieving the highest overall average accuracy of 67.29%. In the base-to-novel setting, TCOC achieves base, novel, and harmonic-mean accuracies of 84.64%, 76.36%, and 80.29%, respectively, outperforming existing category-flexible methods on all three metrics. These results indicate that the text-conditioned generator can construct residuals for new class text features, while orthogonal projection restricts the compensations to directions beyond the principal text-classifier subspace, thereby preserving the original semantic anchor and supporting generalization to new label spaces.

5.4 Ablation and Parameter Analysis

We conduct ablation studies to examine the roles of the two core components. As shown in Table 7, directly optimizing class-wise residuals improves the ImageNet source-domain performance, but the resulting residual matrix is indexed by ImageNet classes and cannot be directly transferred to target datasets with different label spaces. Replacing it with the shared text-conditioned generator enables category-flexible transfer by dynamically producing classifier corrections from target-class text fea-

Table 7: Ablation of components under cross-dataset transfer. All variants are trained on ImageNet1K with 16 shots.

Component	Source	Target Avg.
Class-wise residual	72.13	–
Text-conditioned generator	72.58	65.90
+ Orthogonal compensation	73.12	66.71

Table 8: Geometric analysis. Induced Gap: $\mathcal{G}_{\text{induced}}$, Orthogonal Separation Score: Orth. Sep.

Method	Acc. ↑	Gap ↓	Orth. Sep. ↑
Zero-shot CLIP	66.72	0.7541	–
Text-conditioned generator	72.58	0.7068	0.0055
Random orthogonal residual	67.79	0.8233	0.0138
TCOC	**73.12**	**0.6397**	**0.1235**

Table 9: Parameter analysis on ImageNet1K under the 16-shot setting.

Parameter	β				r_p				r_g			
Value	0.1	0.5	1.0	2.0	0	4	8	16	64	128	256	512
Acc. (%)	71.17	72.89	**73.12**	72.73	72.58	73.01	**73.12**	73.06	71.94	72.39	72.88	**73.12**

tures. Further introducing orthogonal compensation improves the source accuracy to 73.12% and the target-domain average to 66.71%. These results show that text-conditioned generation removes the dependence on fixed class identities, while orthogonal compensation improves both source-task adaptation and cross-dataset transfer. Table 9 analyzes the main hyperparameters of TCOC.

5.5 Geometric Analysis

We further examine whether the learned residuals capture class-discriminative information in the complementary subspace. Under the ImageNet1K 16-shot setting, we measure the overlap between the unprojected residual R and the principal text-classifier subspace: $\rho_{\text{text}} = \frac{\|RB_P\|_F^2}{\|R\|_F^2}$. Without the orthogonal compensation component, **60.82%** of the residual energy lies in the rank-8 principal text-classifier subspace, indicating that the unconstrained generator substantially reuses a small set of dominant text-classifier directions. These directions largely overlap with the original text-classifier information, whereas orthogonal compensation further restricts the residual to the complementary subspace. This also explains why a low-rank projection ($r_p = 8$) is sufficient in practice, as most of the residual energy is already concentrated in these principal text directions.

Table 8 then evaluates the quality of the residual directions after projection. In addition to the induced gap $\mathcal{G}_{\text{induced}}$, we define the orthogonal separation score (Orth. Sep.) to measure class-discriminative alignment in the complementary subspace. Let $\bar{x}$ be the global mean of the test image features and $\mu_c = \frac{1}{N_c}\sum_{i:y_i=c}(x_i - \bar{x})$ be the centered visual prototype of class c. We define $\widehat{\mu}_c^{\perp} = \text{Norm}(\mu_c(I - P_T))$, $\widehat{r}_c^{\perp} = \text{Norm}(r_c(I - P_T))$, where r_c is the c-th row of R, then:

$$\text{OrthSep} = \frac{1}{C}\sum_{c=1}^{C}\left[\widehat{r}_c^{\perp}(\widehat{\mu}_c^{\perp})^{\top} - \frac{1}{C-1}\sum_{k\neq c}\widehat{r}_c^{\perp}(\widehat{\mu}_k^{\perp})^{\top}\right]. \tag{13}$$

A higher score indicates stronger alignment with the corresponding prototype and greater separation from other classes. Variants in Table 8 introduce residual matrices of the same size, controlling the nominal capacity of the expanded classifier space. Random orthogonal residual tests whether arbitrary subspace expansion alone is sufficient. TCOC achieves the lowest induced gap and the highest Orth. Sep., outperforming both the unprojected generator and the random orthogonal baseline. These results show that its residuals exhibit stronger visual alignment in the complementary subspace, thereby mitigating the visual–classifier mismatch. Additional analyses of backbones, reference text-coordinate, and prompt configurations are provided in the **Appendix D**.

6 Conclusion

In this paper, we revisit CLIP-based few-shot adaptation from a modality-gap-inspired perspective and propose TCOC, which generates text-conditioned residuals and projects them beyond the principal text directions. By introducing complementary visual information while preserving the original semantic prior, TCOC achieves effective few-shot adaptation with category flexibility. Future work may explore tighter integration between compensation strategies and PEFT methods.

AI USE STATEMENT

Generative AI tools are used to assist with language polishing and improve the clarity and readability of the manuscript. All AI-assisted text is carefully reviewed and revised by the authors.

REPRODUCIBILITY STATEMENT

We provide the experimental settings and evaluation protocols in the main paper Subsection 5.1, with detailed implementation configurations and hyperparameters documented in the Appendix C. These materials are intended to provide the information necessary to reproduce our results. We will publicly release the source code upon acceptance.

REFERENCES

Yassir Bendou, Amine Ouasfi, Vincent Gripon, and Adnane Boukhayma. Proker: A kernel perspective on few-shot adaptation of large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25092–25102, 2025.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European Conference on Computer Vision*, pp. 446–461. Springer, 2014.

Yichao Cai, Yuhang Liu, Erdun Gao, Tianjiao Jiang, Zhen Zhang, Anton van den Hengel, and Javen Qinfeng Shi. On the value of cross-modal misalignment in multimodal representation learning. In *Advances in Neural Information Processing Systems*, 2025.

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*, pp. 3606–3613, 2014.

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*, pp. 248–255. IEEE, 2009.

Sedigheh Eslami and Gerard de Melo. Mitigate the gap: Improving cross-modal alignment in clip. In *International Conference on Learning Representations*, 2025.

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*, pp. 178–178. IEEE, 2004.

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*, 132(2):581–595, 2024.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

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.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8340–8349, 2021a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15262–15271, 2021b.

Linlan Huang, Xusheng Cao, Haori Lu, Yifan Meng, Fei Yang, and Xialei Liu. Mind the gap: Preserving and compensating for the modality gap in clip-based continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3777–3786, 2025.

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*, pp. 23773–23782, 2024.

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*, pp. 19113–19122, 2023.

Taha Koleilat, Hassan Rivaz, and Yiming Xiao. Clip-svd: Efficient and interpretable vision–language adaptation via singular values. *Transactions on Machine Learning Research*, 2026.

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*, pp. 554–561, 2013.

Yayuan Li, Jintao Guo, Lei Qi, Wenbin Li, and Yinghuan Shi. Text and image are mutually beneficial: Enhancing training-free few-shot classification with clip. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 5039–5047, 2025a.

Yilun Li, Miaomiao Cheng, Xu Han, and Wei Song. Divergence-enhanced knowledge-guided context optimization for visual-language prompt tuning. In *The Thirteenth International Conference on Learning Representations*, 2025b.

Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625, 2022.

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

Marco Mistretta, Alberto Baldrati, Lorenzo Agnolucci, Marco Bertini, and Andrew D. Bagdanov. Cross the gap: Exposing the intra-modal misalignment in clip via modality inversion. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=VVVfuIcmKR.

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

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3498–3505. IEEE, 2012.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

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

Simon Schrodi, David T. Hoffmann, Max Argus, Volker Fischer, and Thomas Brox. Two effects, one trigger: On the modality gap, object bias, and information imbalance in contrastive vision-language models. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=uAFHCZRmXk.

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.

Yuwei Tang, Zhenyi Lin, Qilong Wang, Pengfei Zhu, and Qinghua Hu. Amu-tuning: Effective logit bias for clip-based few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23323–23333, 2024.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. *Advances in neural information processing systems*, 32, 2019.

Zhengbo Wang, Jian Liang, Lijun Sheng, Ran He, Zilei Wang, and Tieniu Tan. A hard-to-beat baseline for training-free clip-based adaptation. *arXiv preprint arXiv:2402.04087*, 2024.

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

Shin'ya Yamaguchi, Dewei Feng, Sekitoshi Kanai, Kazuki Adachi, and Daiki Chijiwa. Post-pre-training for modality alignment in vision-language foundation models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 4256–4266, 2025.

Yuwei Yang, Munawar Hayat, Zhao Jin, Hongyuan Zhu, and Yinjie Lei. Zero-shot point cloud segmentation by semantic-visual aware synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11586–11596, 2023.

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*, pp. 6757–6767, 2023.

Hantao Yao, Rui Zhang, and Changsheng Xu. Tcp: Textual-based class-aware prompt tuning for visual-language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23438–23448, 2024.

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*, pp. 10899–10909, 2023.

Maxime Zanella and Ismail Ben Ayed. Low-rank few-shot adaptation of vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1593–1603, 2024.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512*, 2023.

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*, pp. 493–510. Springer, 2022.

Da-Wei Zhou, Kai-Wen Li, Jingyi Ning, Han-Jia Ye, Lijun Zhang, and De-Chuan Zhan. External knowledge injection for clip-based class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3314–3325, 2025.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16816–16825, 2022a.

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

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*, pp. 15659–15669, 2023a.

Xiangyang Zhu, Renrui Zhang, Bowei He, Aojun Zhou, Dong Wang, Bin Zhao, and Peng Gao. Not all features matter: Enhancing few-shot clip with adaptive prior refinement. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2605–2615, 2023b.

APPENDIX

A THEORETICAL PROOFS AND CLARIFICATION

A.1 PROOF OF PROPOSITION 1

Let $B_P \in \mathbb{R}^{D\times r_p}$ contain orthonormal basis vectors of the principal text-classifier subspace, such that

$$B_P^\top B_P = I_{r_p}, \qquad P_T = B_P B_P^\top.$$

The feasible classifier set is

$$\mathcal{W}_T = \left\{ W_T = LB_P^\top \;\middle|\; L \in \mathbb{R}^{C\times r_p} \right\}.$$

For any $W_T \in \mathcal{W}_T$, we have

$$W_T P_T = LB_P^\top B_P B_P^\top = LB_P^\top = W_T.$$

Therefore, the rows of W_T lie entirely in $\mathrm{span}(B_P)$.

We consider the constrained approximation problem

$$\min_{W_T\in\mathcal{W}_T} \|W_T - W^\star\|_F^2.$$

Since P_T is the orthogonal projector onto $\mathrm{span}(B_P)$, the oracle classifier can be decomposed as

$$W^\star = W^\star P_T + W^\star(I - P_T).$$

For any feasible $W_T \in \mathcal{W}_T$, we thus have

$$W_T - W^\star = (W_T - W^\star P_T) - W^\star(I - P_T).$$

The two terms on the right-hand side are orthogonal under the Frobenius inner product, because the rows of the first term lie in $\mathrm{span}(B_P)$ and those of the second term lie in $\mathrm{span}(B_P)^\perp$. Hence,

$$\|W_T - W^\star\|_F^2 = \|W_T - W^\star P_T\|_F^2 + \|W^\star(I - P_T)\|_F^2.$$

The second term is independent of W_T, whereas the first term is nonnegative and is minimized if and only if

$$W_T = W^\star P_T.$$

Therefore, the best approximation to $W^\star$ within $\mathcal{W}_T$ is

$$\widehat{W_T} = W^\star P_T,$$

with approximation error

$$\mathcal{E}_T = \min_{W_T\in\mathcal{W}_T} \|W_T - W^\star\|_F^2 = \|W^\star(I - P_T)\|_F^2.$$

A.2 PROOF OF COROLLARY 1

Let

$$\mathcal{S}_T = \mathrm{span}(B_P), \qquad P_T = B_P B_P^\top,$$

and let P_R be the orthogonal projector onto a residual subspace

$$\mathcal{S}_R \subseteq \mathcal{S}_T^\perp.$$

Since $\mathcal{S}_T$ and $\mathcal{S}_R$ are orthogonal, their projectors satisfy

$$P_T P_R = P_R P_T = 0.$$

Consequently,

$$P_\oplus = P_T + P_R$$

is the orthogonal projector onto the direct-sum subspace

$$\mathcal{S}_T \oplus \mathcal{S}_R.$$

Define the corresponding feasible classifier set as

$$\mathcal{W}_{\oplus} = \left\{ W \in \mathbb{R}^{C \times D} \;\middle|\; \mathrm{row}(W) \subseteq \mathcal{S}_T \oplus \mathcal{S}_R \right\}.$$

For every $W \in \mathcal{W}_{\oplus}$, we have

$$W = W(P_T + P_R).$$

We consider the constrained approximation problem

$$\min_{W \in \mathcal{W}_{\oplus}} \|W - W^\star\|_F^2.$$

Using the orthogonal projector $P_{\oplus} = P_T + P_R$, the oracle classifier can be decomposed as

$$W^\star = W^\star(P_T + P_R) + W^\star(I - P_T - P_R).$$

Therefore, for any $W \in \mathcal{W}_{\oplus}$,

$$W - W^\star = [W - W^\star(P_T + P_R)] - W^\star(I - P_T - P_R).$$

The two terms are orthogonal under the Frobenius inner product. It follows that

$$\begin{aligned} &\|W - W^\star\|_F^2 \\ &= \|W - W^\star(P_T + P_R)\|_F^2 + \|W^\star(I - P_T - P_R)\|_F^2 . \end{aligned}$$

The first term is minimized if and only if

$$W = W^\star(P_T + P_R).$$

Hence, the best approximation in the direct-sum subspace is

$$W_{\oplus}^\star = W^\star(P_T + P_R),$$

with approximation error

$$\mathcal{E}_{\oplus} = \|W^\star(I - P_T - P_R)\|_F^2 .$$

To compare this error with the text-subspace approximation error, observe that

$$I - P_T = P_R + (I - P_T - P_R).$$

Thus,

$$W^\star(I - P_T) = W^\star P_R + W^\star(I - P_T - P_R).$$

The two components on the right-hand side are orthogonal because P_R projects onto $\mathcal{S}_R$, whereas $I - P_T - P_R$ projects onto the orthogonal complement of $\mathcal{S}_T \oplus \mathcal{S}_R$. Therefore,

$$\|W^\star(I - P_T)\|_F^2 = \|W^\star P_R\|_F^2 + \|W^\star(I - P_T - P_R)\|_F^2.$$

Using

$$\mathcal{E}_T = \|W^\star(I - P_T)\|_F^2$$

and

$$\mathcal{E}_{\oplus} = \|W^\star(I - P_T - P_R)\|_F^2,$$

we obtain

$$\mathcal{E}_T - \mathcal{E}_{\oplus} = \|W^\star P_R\|_F^2.$$

Since $P_T P_R = 0$, we have

$$(I - P_T)P_R = P_R,$$

and hence

$$\mathcal{E}_T - \mathcal{E}_{\oplus} = \|W^\star(I - P_T)P_R\|_F^2 \geq 0.$$

Therefore,

$$\mathcal{E}_{\oplus} \leq \mathcal{E}_T.$$

A.3 THEORY-TO-PRACTICE CLARIFICATION

The above results provide an approximation-based geometric motivation for orthogonal compensation rather than an exact characterization of the optimization procedure used by TCOC. The oracle classifier $W^\star$ is introduced only for analysis and is not available during training. Proposition and Corollary show that, when classifier directions are restricted to a principal text-classifier subspace, components outside this subspace cannot be represented, whereas adding an orthogonal residual subspace enlarges the feasible classifier space and cannot increase the best linear approximation error.

In practice, TCOC does not explicitly optimize over all classifiers in an arbitrary residual subspace. Instead, it generates a structured residual from class text coordinates,

$$R = h_\theta(TB_G),$$

and projects it as

$$R_\perp = R(I - P_T).$$

Therefore, every row of $R_\perp$ lies in the orthogonal complement of the protected principal text-classifier subspace. The final classifier is constructed as

$$W = \mathrm{Norm}(T + \beta R_\perp).$$

The theory does not claim that the learned residual exactly recovers the optimal residual subspace P_R or achieves the oracle approximation error. Rather, it explains why restricting the learned correction to complementary directions is geometrically well motivated when the principal text-classifier subspace does not fully represent the visual discriminative structure.

B ALGORITHM AND ARCHITECTURE DETAILS

B.1 BASIS CONSTRUCTION AND NOTATION

TCOC uses two text-derived bases with different purposes: the text-coordinate basis B_G, which provides a shared input coordinate system for the residual generator, and the projection basis B_P, which defines the principal text-classifier subspace protected during orthogonal compensation. We detail their construction below.

Text-coordinate basis. Let

$$T_{\mathrm{ref}} = [t_1^{\mathrm{ref}}, \ldots, t_{C_{\mathrm{ref}}}^{\mathrm{ref}}]^\top \in \mathbb{R}^{C_{\mathrm{ref}} \times D}$$

denote the row-normalized text classifier constructed from a reference label set. We perform compact SVD:

$$T_{\mathrm{ref}} = U_G \Sigma_G V_G^\top,$$

where

$$U_G \in \mathbb{R}^{C_{\mathrm{ref}} \times q_G}, \quad \Sigma_G \in \mathbb{R}^{q_G \times q_G}, \quad V_G \in \mathbb{R}^{D \times q_G},$$

and

$$q_G = \min\{C_{\mathrm{ref}}, D\}.$$

The text-coordinate basis is formed by the leading r_g right singular vectors:

$$B_G = V_G[:, 1 : r_g] \in \mathbb{R}^{D \times r_g}, \qquad r_g = \min\{C_{\mathrm{ref}}, D, d_g\},$$

where d_g is the predefined maximum coordinate dimension.

Given a text classifier

$$T \in \mathbb{R}^{C \times D}$$

for the current label space, its class-wise coordinates in the reference text system are

$$U = TB_G \in \mathbb{R}^{C \times r_g}.$$

The shared residual generator is then applied independently to each row of U:

$$R = h_\theta(U) \in \mathbb{R}^{C \times D}.$$

By default, the reference label set is the few-shot training label set, such that $T_{\mathrm{ref}} = T$ during standard few-shot training. Once constructed, B_G remains fixed throughout training and inference. In particular, when TCOC is transferred to a new label space, the new class text features are represented using the same coordinate basis B_G. This provides a common input coordinate system for the learned generator across label spaces with different class numbers and semantics.

Projection basis. For the text classifier associated with the current prediction label space, we perform compact SVD:

$$T = U_P \Sigma_P V_P^\top,$$

where

$$U_P \in \mathbb{R}^{C \times q_P}, \quad \Sigma_P \in \mathbb{R}^{q_P \times q_P}, \quad V_P \in \mathbb{R}^{D \times q_P},$$

and

$$q_P = \min\{C, D\}.$$

The projection basis is constructed from the leading r_p right singular vectors:

$$B_P = V_P[:, 1 : r_p] \in \mathbb{R}^{D \times r_p}.$$

Since the columns of B_P are orthonormal, the orthogonal projector onto the protected principal text-classifier subspace is

$$P_T = B_P B_P^\top \in \mathbb{R}^{D \times D}.$$

The generated residual is projected onto its orthogonal complement as

$$R_\perp = R(I - P_T) = R - (RB_P)B_P^\top.$$

The latter form is used in implementation and avoids explicitly constructing the $D \times D$ identity and projection matrices.

Unlike B_G, the projection basis depends on the label space used for prediction. During standard few-shot training and inference, B_P is computed from the training-class text classifier. When the inference label space changes from $\mathcal{Y}$ to $\mathcal{Y}'$, we first reconstruct the corresponding text classifier

$$T' \in \mathbb{R}^{C' \times D}$$

from the new class names and then recompute its projection basis:

$$T' = U'_P \Sigma'_P V_P'^\top, \qquad B'_P = V'_P[:, 1 : r_p].$$

The inference-time residual therefore satisfies

$$R'_\perp B'_P = 0,$$

ensuring that the compensation remains orthogonal to the principal text directions of the new label space.

In summary, B_G defines a fixed reference coordinate system for category-flexible residual generation, whereas B_P defines a label-space-dependent protected subspace for orthogonal compensation. During transfer, B_G and h_θ remain fixed, while the text classifier and its projection basis are reconstructed from the inference class names.

B.2 Training Procedure

See Pseudocode 1.

B.3 Category-Flexible Inference

See Pseudocode 2.

B.4 Residual Generator Architecture

The residual generator is a lightweight row-wise mapping shared across all classes. Given the text coordinates $U = TB_G \in \mathbb{R}^{C \times r_g}$, it consists of a linear branch and a two-layer MLP branch:

$$h_\theta(U) = UW_{\text{lin}} + \sigma(UW_1 + b_1)W_2,$$

where

$$W_{\text{lin}} \in \mathbb{R}^{r_g \times D}, \quad W_1 \in \mathbb{R}^{r_g \times d_h}, \quad W_2 \in \mathbb{R}^{d_h \times D},$$

$\sigma(\cdot)$ is ReLU, and the hidden dimension is $d_h = 128$. The generator is independently applied to each class row and contains no class-indexed parameters, so its parameter size is independent of the number of classes. Both W_{lin} and W_2 are zero-initialized, ensuring that the initial residual is zero and that training starts from the original CLIP text classifier.

Algorithm 1 Training Procedure of TCOC

Input: Few-shot support set $\mathcal{D}_s$; frozen encoders f and g; training label set $\mathcal{Y}$; reference text classifier T_{ref}; ranks r_g and r_p; compensation strength β
Output: Residual generator h_θ and text-coordinate basis B_G

Construct the normalized text classifier $T \in \mathbb{R}^{C\times D}$ from $\mathcal{Y}$
Compute B_G from the top-r_g right singular vectors of T_{ref} ($T_{\text{ref}} = T$ by default)
Compute B_P from the top-r_p right singular vectors of T
Extract and normalize the support image features
for each training epoch **do**
  for each mini-batch (X_b, Y_b) from $\mathcal{D}_s$ **do**
    $U \leftarrow TB_G$
    $R \leftarrow h_\theta(U)$
    $R_\perp \leftarrow R - (RB_P)B_P^\top$
    $W \leftarrow \mathrm{Norm}(T + \beta R_\perp)$
    $Z_b \leftarrow \tau X_b W^\top$
    $\mathcal{L} \leftarrow \mathrm{CE}(Z_b, Y_b)$
    Update θ by minimizing $\mathcal{L}$
  end for
end for
return h_θ and B_G

Algorithm 2 Category-Flexible Inference of TCOC

Input: Test image I; inference label set $\mathcal{Y}'$; frozen encoders f and g; learned h_θ and B_G; projection rank r_p; compensation strength β
Output: Predicted class $\hat{y}$

Construct the normalized text classifier $T' \in \mathbb{R}^{C'\times D}$ from $\mathcal{Y}'$
Compute B'_P from the top-r_p right singular vectors of T'
$U' \leftarrow T'B_G$
$R' \leftarrow h_\theta(U')$
$R'_\perp \leftarrow R' - (R'B'_P)B_P'^\top$
$W' \leftarrow \mathrm{Norm}(T' + \beta R'_\perp)$
$x \leftarrow \mathrm{Norm}(f(I))$
$z \leftarrow \tau x W'^\top$
$\hat{y} \leftarrow \arg\max_{c\in\mathcal{Y}'} z_c$
return $\hat{y}$

C Experimental and Reproducibility Details

C.1 Evaluation Protocols

The datasets and evaluation settings are described in the main paper. Here, we provide several additional details regarding label-space changes and metric computation. All standard few-shot experiments follow the dataset splits introduced by CoOp (Zhou et al., 2022b) and are conducted under $K \in \{1, 2, 4, 8, 16\}$ shots.

For domain generalization, the model is trained only on ImageNet1K under the 16-shot setting and directly evaluated on ImageNet-V2, ImageNet-Sketch, ImageNet-A, and ImageNet-R. Since these datasets preserve the ImageNet label semantics, the same class set is used during inference.

For cross-dataset transfer, the model is also trained only on ImageNet1K with 16 shots and is directly transferred to the other ten datasets. No target-domain image is used for training, model selection, or hyperparameter tuning. For each target dataset, we only reconstruct the text classifier and its projection basis from the target class names, while keeping the learned text-coordinate basis and residual generator fixed.

For base-to-novel generalization, the model is trained only using few-shot samples from the base classes and is evaluated separately on the base and novel classes. During novel-class inference, the

Table 10: Prompt templates used to construct the ensemble text classifier. ImageNet uses seven templates following common CLIP few-shot adaptation settings, while the other ten datasets use five dataset-specific templates generated with GPT-3.

Dataset	Prompt Templates
ImageNet	`itap of a {}.; a bad photo of the {}.; a origami {}.; a photo of the large {}.; a {} in a video game.; art of the {}.; a photo of the small {}.`
Caltech101	`a photo of a {}.; a close-up photo of a {}.; a photo of the object {}.;a cropped photo of a {}.;a clean photo of a {}.`
OxfordPets	`a photo of a {}, a type of pet.; a photo of the pet {}.; a close-up photo of a {}.; a photo of a {} animal.; a portrait photo of a {}.`
StanfordCars	`a photo of a {}.; a photo of the car {}.; a side view photo of a {}.;a photo of a {} vehicle.;a close-up photo of a {}.`
Flowers102	`a photo of a {}, a type of flower.; a close-up photo of a {} flower.; a photo of the flower {}.; a macro photo of a {}.; a photo of a blooming {}.`
Food101	`a photo of {}, a type of food.; a close-up photo of {}.; a photo of a plate of {}.; a photo of freshly made {}.; a restaurant photo of {}.`
FGVCAircraft	`a photo of a {}, a type of aircraft.; a photo of the aircraft {}.; a side view photo of a {}.; a photo of a {} airplane.;a photo of a flying {}.`
SUN397	`a photo of a {}.; a photo of the scene {}.; a photo of a {} scene.;an indoor or outdoor scene of {}.;a view of a {}.`
DTD	`{} texture.; a photo of {} texture.; a close-up photo of {} texture.; a photo showing a {} surface.; a texture image of {}.`
EuroSAT	`a centered satellite photo of {}.; a satellite image of {}.; an aerial photo of {}.; a remote sensing image of {}.; a satellite view of {}.`
UCF101	`a photo of a person doing {}.; a video frame of a person doing {}.; a photo of the action {}.; a person is performing {}.;a scene of a person doing {}.`

novel-class text classifier and the corresponding projection basis are reconstructed from the novel class names, without using novel-class images for adaptation. In addition to the base and novel accuracies, we report their harmonic mean:

$$\mathrm{HM} = \frac{2A_{\mathrm{base}}A_{\mathrm{novel}}}{A_{\mathrm{base}} + A_{\mathrm{novel}}},$$

where A_{base} and A_{novel} denote the classification accuracies on the base and novel classes, respectively.

C.2 Implementation Details and Prompt Ensembles

Unless otherwise specified, our implementation follows the training configuration of TaskRes (Yu et al., 2023). In particular, we adopt the same Adam optimizer configuration, learning-rate warm-up strategy, numerical precision, and image preprocessing pipeline. The CLIP image and text encoders remain frozen throughout training, and only the text-conditioned residual generator is optimized. We use the checkpoint from the final training epoch for evaluation.

To obtain stable and semantically informative text classifiers, we use prompt ensembles for all datasets. For ImageNet, we adopt seven prompt templates following the settings commonly used by TaskRes (Yu et al., 2023) and Tip-Adapter (Zhang et al., 2022). For the other ten datasets, following TIMO (Li et al., 2025a), we use five dataset-specific descriptions generated with GPT-3, as shown in Table 10. These prompt sets are generated once and remain fixed across all shot settings, random seeds, and evaluation protocols.

Let $p_m(c)$ denote the m-th prompt instantiated with class name c. We first normalize the text feature produced by each individual prompt, average the normalized features, and normalize the resulting class representation again:

$$t_c = \mathrm{Norm}\left(\frac{1}{M}\sum_{m=1}^{M}\mathrm{Norm}\big(g(p_m(c))\big)\right),$$

where M denotes the number of prompt templates and $g(\cdot)$ is the CLIP text encoder. The resulting class representations are stacked row-wise to construct the text classifier T.

Prompt ensembling is particularly relevant to TCOC because the text representations are used not only as the original classifier weights, but also to construct the text-coordinate basis B_G and the projection basis B_P. Compared with a single prompt, an ensemble of complementary descriptions provides a more complete and stable representation of class semantics, which in turn affects both the input coordinates of the residual generator and the principal text directions protected by orthogonal compensation. Additional experiments in the appendix examine the interaction between prompt ensembling and TCOC and show that richer text representations provide further benefits for basis construction and compensation learning.

C.3 Hyperparameter Selection

We follow the validation protocol of CoOp for hyperparameter selection. Specifically, the learning rate is selected on the validation split of the corresponding dataset from

$$\left\{10^{-3}, 5\times10^{-4}, 10^{-4}\right\}.$$

The selected configuration is then used for final evaluation on the test split. The number of training epochs, batch size, compensation strength β, text-coordinate rank r_g, and projection rank r_p, together with their final settings, are provided in the main paper. For domain generalization and cross-dataset transfer, we use the same ImageNet1K 16-shot checkpoint as the standard ImageNet few-shot experiment. This checkpoint is fixed without retraining. In these generalization experiments, we consistently set the projection rank to $r_p = 8$ for all evaluation settings. For base-to-novel generalization, a separate model is trained on the base classes of each dataset and evaluated on its corresponding base and novel classes. No target-domain or novel-class image is used for hyperparameter tuning. In particular, we use $\beta_{\mathrm{train}} = 1.0$ throughout source-domain training. At inference time, we heuristically reduce the residual contribution to retain a stronger influence from the original text classifier:

$$\beta_{\mathrm{test}} = \begin{cases} 1.0, & \text{standard few-shot classification,} \\ 0.5, & \text{domain and cross-dataset generalization,} \\ 0.2, & \text{base-to-novel generalization.} \end{cases}$$

Each value is fixed for the entire evaluation protocol and is shared by all corresponding target datasets. We do not perform dataset-specific tuning using target test results.

Following prior studies such as CLIP-SVD (Koleilat et al., 2026), we adopt these protocol-specific adaptation strengths for different generalization settings. Unlike CLIP-SVD, which further adjusts the source-domain training configuration to preserve transferability, **TCOC keeps all training settings unchanged, including the training objective, schedule, and $\beta_{\mathrm{train}} = 1.0$, and only adjusts the residual contribution at inference time.** In particular, domain generalization and cross-dataset transfer use the same ImageNet1K source-trained checkpoint, without retraining or target-specific checkpoint selection. The inference-time value is fixed at the protocol level and shared by all datasets within that protocol.

C.4 Randomness and Computing Environment

All main experiments are conducted using three independent random seeds: $\{1, 2, 3\}$. Unless otherwise specified, all reported TCOC results are the arithmetic mean over these three independent runs. All experiments are performed on a single NVIDIA H200 GPU.

Table 11: Few-shot comparison on 11 datasets using ResNet-50 CLIP. Several competing methods use label-space-specific caches or classifiers, whereas TCOC preserves category flexibility.

Method	Shots					Avg.
	1	2	4	8	16	
Zero-shot CLIP	58.77	58.77	58.77	58.77	58.77	58.77
CoOp	59.59	62.32	66.77	69.89	73.42	66.40
Tip-Adapter	62.83	64.84	66.50	68.64	70.02	66.56
CLIP-Adapter	62.67	65.55	68.61	71.40	74.44	68.53
GDA	62.19	66.19	69.77	73.30	76.04	69.49
TaskRes	64.04	67.02	69.61	72.76	75.10	69.71
ProKeR	64.01	67.31	70.42	73.85	76.75	70.46
TCOC (Ours)	**64.70**	**67.56**	**70.74**	**74.08**	**76.93**	**70.80**

Table 12: Effect of the reference vocabulary used to construct B_G on few-shot classification with ViT-B/16 CLIP. In the task-specific setting, B_G is constructed separately from the class texts of each dataset. The external-reference setting instead uses a shared basis derived from ImageNet-21K.

Reference for B_G	1	2	4	8	16	Avg.
ImageNet-21K	71.15	73.92	76.53	78.86	81.31	76.35
Task-specific classes	**72.00**	**74.34**	**76.85**	**79.87**	**82.09**	**77.03**

D Additional Experimental Results

D.1 Generalization across Backbones

We further evaluate TCOC with the ResNet-50 (He et al., 2016) CLIP backbone to examine whether its effectiveness is specific to ViT-B/16. As shown in Table 11, TCOC achieves the best accuracy under all shot settings and obtains an average accuracy of 70.80%. Notably, several competing methods in this comparison rely on class-indexed caches, residuals, or classifiers and are therefore tied to the training label space. In contrast, TCOC achieves stronger performance without introducing class-specific parameters. Together with the ViT-B/16 results in the main paper, these results indicate that TCOC remains effective across both convolutional and Transformer-based CLIP backbones.

D.2 Alternative Reference Label Sets

We examine whether the text-coordinate basis B_G must be constructed from the label set of the current training task. For standard few-shot classification, the default TCOC constructs B_G separately from the class texts of each dataset. We compare this task-specific construction with a shared external basis derived from the ImageNet-21K vocabulary. For cross-dataset transfer, the default reference is the ImageNet1K source label set, and we similarly replace it with ImageNet-21K.

In all comparisons, only the reference vocabulary used to construct B_G is changed. The task text classifier T, the label-space-dependent projection basis B_P, the residual-generator architecture, and all remaining training settings are kept unchanged. Since replacing B_G changes the input coordinate system, the residual generator is retrained for each reference setting.

As shown in Table 12, constructing B_G from the class texts of the current dataset achieves the strongest overall few-shot performance. Compared with the shared ImageNet-21K basis, the task-specific basis improves the average accuracy from 76.35% to 77.03%. This result is reasonable because the task-specific vocabulary provides a coordinate system more closely aligned with the semantic structure of the current label space. Nevertheless, the external ImageNet-21K basis remains consistently effective across all shot settings, showing that TCOC does not require B_G to be derived from the current task classes.

A similar trend is observed in cross-dataset transfer, as shown in Table 13. The ImageNet1K source reference achieves a target average of 66.71%, compared with 66.26% for ImageNet-21K, while the latter performs slightly better on several individual target datasets. These results indicate that a

Table 13: Effect of the reference vocabulary under cross-dataset transfer. TCOC is trained on ImageNet1K with 16 shots and directly evaluated on the other ten datasets. The default setting constructs B_G from the ImageNet1K source class texts, whereas the external-reference setting uses ImageNet-21K. Tgt. Avg. denotes the average over the ten target datasets, and All Avg. additionally includes the ImageNet source result.

Reference	IN	Tgt. Avg.	All Avg.	Caltech	Pets	Cars	Flowers	Food	Aircraft	SUN	DTD	EuroSAT	UCF
IN-21K	73.08	66.26	66.88	**94.16**	**90.02**	**65.99**	70.85	86.54	**25.68**	66.22	44.50	50.26	68.38
IN-1K	**73.12**	**66.71**	**67.29**	94.05	89.91	65.98	**71.25**	**86.81**	25.55	**67.23**	**45.21**	**52.30**	**68.80**

Table 14: Effect of prompt ensembling on few-shot classification with ViT-B/16 CLIP. We report the average classification accuracy (%) over 11 datasets. Gain denotes the improvement obtained by replacing a single prompt with the prompt ensemble.

Method	w/o Prompt Ensemble	w/ Prompt Ensemble	Gain
CLIP (zero-shot)	65.36	65.85	+0.49
TCOC (1-shot)	71.04	72.00	+0.96
TCOC (2-shot)	73.75	74.34	+0.59
TCOC (4-shot)	76.26	76.85	+0.59
TCOC (8-shot)	78.85	79.87	+1.02
TCOC (16-shot)	81.35	82.09	+0.74
TCOC (Avg. over shots)	76.25	77.03	+0.78

broad external vocabulary can provide a transferable shared coordinate system, although increasing the vocabulary size does not necessarily improve aggregate accuracy. A reference set that is better aligned with the source task may produce more compact and relevant text coordinates. We therefore use the task-specific label set for standard few-shot adaptation and the ImageNet1K source labels for cross-dataset transfer as our default settings.

D.3 Effect of Prompt Ensembles

As shown in Table 14, prompt ensembling consistently improves both zero-shot CLIP and TCOC. However, the average gain of TCOC reaches 0.78%, exceeding the 0.49% improvement obtained by the original CLIP classifier. This suggests that TCOC benefits from prompt ensembling not only through a stronger text classifier. Since the ensemble text features are also used to construct the text-coordinate basis B_G and the projection basis B_P, aggregating multiple descriptions reduces the sensitivity to a particular prompt and yields a more stable representation of the class semantics. Consequently, both the coordinates provided to the residual generator and the principal text directions protected by orthogonal compensation become more reliable. The consistently larger gains of TCOC therefore indicate a positive coupling between richer prompt representations and the proposed compensation mechanism.

D.4 Comparison under Matched Prompt Ensembles

The main experiments use prompt ensembles to construct the text classifiers of TCOC. To examine whether the improvement of TCOC mainly results from access to richer textual descriptions, we additionally retrain TCP, the strongest category-flexible competitor in our main comparison, using exactly the same prompt ensembles. Specifically, we replace the single-template class text features used by TCP for class-aware conditioning and semantic regularization with the normalized ensemble of the same dataset-specific templates. The original learnable prompt architecture, optimization objective, and training configuration of TCP remain unchanged. For cross-dataset transfer, TCP is trained on ImageNet1K, and the class text features of each target dataset are reconstructed using the corresponding target prompt ensemble, without using any target-domain image for adaptation or model selection.

Table 15: Few-shot comparison under matched prompt ensembles. For TCP w/ Ensemble, the fixed class text features are constructed using the same dataset-specific prompt ensembles as TCOC, while the original learnable prompt architecture and training configuration remain unchanged.

Method	1	2	4	8	16	Avg.
TCP	70.63	73.59	76.07	78.39	80.98	75.93
TCP w/ Ensemble	70.50	73.35	76.09	78.89	80.93	75.95
TCOC (Ours)	**72.00**	**74.34**	**76.85**	**79.87**	**82.09**	**77.03**

Table 16: Cross-dataset transfer comparison under matched prompt ensembles. All methods are trained on ImageNet1K with 16 shots and directly evaluated on the other ten datasets. For TCP w/ Ensemble, both the source class text features and the target class text features are constructed using the same dataset-specific prompt ensembles as TCOC.

	Source	Average		Target									
Method	IN	Tgt.	All	Caltech	Pets	Cars	Flowers	Food	Aircraft	SUN	DTD	EuroSAT	UCF
TCP	71.40	66.29	66.76	93.97	**91.25**	64.69	71.21	86.69	23.45	67.15	44.35	51.45	68.73
TCP w/ Ensemble	72.13	65.41	66.02	93.91	89.13	64.39	71.13	86.04	24.42	66.15	44.50	48.81	65.58
TCOC (Ours)	**73.12**	**66.71**	**67.29**	**94.05**	89.91	**65.98**	**71.25**	**86.81**	**25.55**	**67.23**	**45.21**	**52.30**	**68.80**

Table 17: Effect of the inference-time projection basis under cross-dataset transfer. TCOC is trained once on ImageNet1K with 16 shots and evaluated on the other datasets. The source variant retains the ImageNet projection basis, whereas the target variant recomputes the basis using the target class names. ρ_{tgt} measures the average residual-energy ratio within the principal target text subspace.

Projection Basis	Caltech	Pets	Cars	Flowers	Food	Aircraft	SUN	DTD	EuroSAT	UCF	Avg.	ρ_{tgt} (%) $\downarrow$
Source B_P^{src}	94.20	88.58	65.84	70.89	86.24	25.56	66.20	44.44	50.68	68.12	66.08	2.58
Target B_P^{tgt}	94.05	89.91	65.98	71.25	86.81	25.55	67.23	45.21	52.30	68.80	66.71	0.00

As shown in Table 15, introducing the matched prompt ensemble changes the average few-shot accuracy of TCP only marginally, from 75.93% to 75.95%, whereas TCOC achieves 77.03%. A similar observation holds under cross-dataset transfer. As reported in Table 16, the matched ensemble does not improve the transfer performance of TCP: its target-dataset average changes from 66.29% to 65.41%, and its overall average changes from 66.76% to 66.02%. In contrast, TCOC obtains target and overall averages of 66.71% and 67.29%, respectively. These controlled results indicate that the advantage of TCOC cannot be explained merely by the use of richer prompt templates.

One possible explanation is that TCP already learns adaptive context representations, and therefore replacing its fixed textual prior with an ensemble provides limited additional benefit and may interact nontrivially with the source-trained prompts during label-space transfer. In contrast, TCOC performs classifier-side adaptation and uses the ensemble text representations to construct the original semantic classifier, the text-coordinate basis, and the protected principal text subspace. The richer text representations can provide more stable semantic coordinates and principal directions for residual generation and orthogonal compensation. These suggest that the effect of prompt ensembling is method-dependent, rather than being a generic source of the performance gain of TCOC.

D.5 Inference-Time Projection Basis

As shown in Table 17, recomputing the projection basis from the target text classifier slightly improves the average cross-dataset accuracy from 66.08% to 66.71%. More importantly, the residual overlap with the principal target text subspace is reduced from 2.58% to nearly zero. This confirms that retaining the source ImageNet basis does not strictly preserve orthogonality after the label space changes, whereas the target-specific basis restores the intended geometric constraint using only the target class names. The comparable or improved accuracy further shows that this recomputation does not sacrifice transfer performance. We therefore adopt the inference-time text classifier to reconstruct the projection basis for new label spaces.