A. Overview of Appendix

In addition to the main method and experiments outlined in the paper, we offer supplementary information about our work in the Appendix. In Appendix B, we delve into implementation within the methodology section. We provide more details about our implementation in the Static Anchor and Stochastic Anchor, as well as the Maximum Mean Discrepancy Minimization. Subsequently in Appendix C, we provide more results on dataset setting and adaptation experiments, covering the Base-to-Novel Generalization and group Robustness task. Further in Appendix D, we present additional results of ablation studies along with their visualization outcomes. Lastly in Appendix E and F, we explore the limitations of our work and analyze its broader impact.

B. Method

B.1. Review the Adaption of CLIP

We first review the pretraining and inference stage of the CLIP model, then we discuss the adaptation of CLIP. During the pretraining phase, a large-scale dataset of image-text pairs (x, y) is collected for training the model using a contrastive learning approach. Here x represents an image, and y denotes its corresponding textual description. For each image x, an image encoder model f_θ is parameterized by θ to extract its visual feature vector $u \in \mathbb{R}^{1\times H}$: $u = f_\theta(x)$. Similarly, for each textual description y, a text encoder g_ϕ is parameterized by ϕ to get its feature embedding $v \in \mathbb{R}^{1\times H}$: $v = g_\phi(y)$. For the i-th image x_i and the j-th language description y_j in a batch $\mathcal{B}$, we normalize their feature vectors to a hyper-sphere using:$u_i = \frac{f_\theta(x_i)}{\|f_\theta(x_i)\|}$ and $v_j = \frac{g_\phi(y_j)}{\|g_\phi(y_j)\|}$.

Test phase of CLIP. In this phase, a predefined prompt "a photo of a " is commonly employed for inference. Let's consider a single test image x_{test} of class C, where $x_{\text{test}} \in \mathbb{R}^{C\times H\times W}$ and $C \in \mathbb{R}^K$ for a K-class classification problem. The predefined prompt is prepended to each class label in C to construct the language description. The zero-shot prediction probability for the test image is determined by:

$$Pr\left(c = k \mid x_{\text{test}}\right) = \frac{\exp\left(\operatorname{sim}\left(u, v_i\right)\tau\right)}{\sum_{i=1}^{K}\exp\left(\operatorname{sim}\left(u, v_i\right)\tau\right)} \tag{1}$$

B.2. Introduction of the Prompt Tuning Method

Language Prompt Tuning involves introducing learnable parameters into the text branch. We follow the same notation in [?] and [?]. In the text branch, the class label c is formatted as a language description within a text template as “a photo of a {label}", which can be further transferred as $\tilde{y} = \{t_{SOS}, t_1, t_2, \cdots, t_L, c_n, t_{EOS}\}$. Here t_l are the word embeddings of the text template, and c_n are the class label. The t_{SOS} and t_{EOS} are the learnable start and end token embeddings. The text encoder g encodes the input tokens $\tilde{y}$ through multiple transformer blocks to generate a latent text feature representation $\tilde{g} = g\left(\tilde{y}, \theta_g\right)$. In Language prompt tuning, learnable text prompts $\Lambda_{\text{text}} \in \left\{\lambda_{\text{text}}^1, \lambda_{\text{text}}^2, \cdots, \lambda_{\text{text}}^L\right\}$ are appended to the input $\tilde{y}$. In CoOp [?] or CoCoOp [?], the learnable text prompts λ_{text} are only added to input of text encoder. While in Maple [?], the learnable text prompts are appended to multiple transformer layers as $[\ldots, \tilde{y}_i] = \mathcal{L}_i\left([\Lambda_{\text{text}}, \tilde{y}_{i-1}]\right) \quad i = 1, 2, \cdots, J$, where $\mathcal{L}_i$ represent the i layer number in the transformer, J represent the prompt depth.

Visual Prompt Learning involves the integration of learnable prompts within the image branch. The input image x is divided into M patches, and these patches are projected to generate patch embeddings $\tilde{x} = \{e_{cls}, e_1, e_2, \cdots, e_M\}$, where e_{cls} is the learnable class token. Subsequently, learnable visual prompts are introduced as $\Lambda_{\text{visual}} \in \left\{\lambda_{\text{visual}}^1, \lambda_{\text{visual}}^2, \cdots, \lambda_{\text{visual}}^L\right\}$. The learnable visual prompts are appended to multiple transformer layers as $[c_i, \tilde{x}_i, \ldots] = \mathcal{V}_i\left([c_{i-1}, \tilde{x}_{i-1}, \Lambda_{\text{visual}}]\right) \quad i = 1, 2, \cdots, J$, where $\mathcal{V}_i$ represent the i layer in vision transformer, J represent the prompt depth.

Multi-modal Prompt Learning integrates language prompt learning and visual prompt learning, combining them synergistically. Simply combining text prompt learning and visual prompt learning is called independent prompt learning, which is used in UPT [?]. To foster interaction between the image and text branches, multi-modal prompt learning employs projection layers $L_t = \left\{\tilde{l^1}, \tilde{l^2}, \cdots, \tilde{l^J}\right\}$ for projecting the learnable language prompts onto the visual prompts, defined as $\Lambda_{\text{visual}} = \left\{\tilde{l^1}(\lambda_{\text{text}}^1), \tilde{l^2}(\lambda_{\text{text}}^2), \cdots, \tilde{l^J}(\lambda_{\text{text}}^L)\right\}$. This formulation facilitates interaction between the visual and language prompts. Such unified prompt tuning is a key feature of the Maple [?] framework.

B.3. Static Anchor Implementation

To address the overfitting issues of text-based cross-entropy loss in scenarios with limited data, we propose a symmetrical static anchor alignment method, analogous to an image-based cross-entropy loss. This method involves two primary steps:

Step 1: Construction of Image Anchors. We use a pre-trained CLIP image encoder to extract features for each category in the source dataset, followed by K-means clustering to identify the centroid of each category's features, denoted as a_x^k. It is important to note that the dimensionality of a_x^k differs from that of the batch image features $f_\theta(x)$.

Step 2: Alignment with Text Samples. For each image batch, corresponding text labels represented as language descriptions are aligned, with batch text embeddings $v' = g_\phi(y, \Lambda_{\text{txt}})$, where $v' \in \mathbb{R}^{B\times H}$, also differing in feature dimensions from class labels.

B.4. Stochastic Anchors Implementation

Stochastic anchors, selected during each batch, can be implemented as cross-modal contrastive learning process. Traditional supervised learning, which models relationships between images and discrete labels, often neglects textual concepts associated with labels. In contrast, stochastic anchors learning fosters understanding of visual concepts through enforcing batch-level text-image alignment.

We construct a contrastive similarity matrix $s' = \mathrm{sim}(u, v')$, where $s' \in \mathbb{R}^{N \times N}$. This matrix supports the formulation of both image-to-text and text-to-image contrastive losses, averaging these to derive the final text-image contrastive loss. In this matrix, only diagonal entries are treated as positive examples, enhancing the robustness of the latent space by introducing a larger set of negative samples.

B.5. Maximum Mean Discrepancy Implementation

Maximum Mean Discrepancy (MMD) is a kernel-based method primarily used to test the equality of two distributions from samples. Introduced in [**?**], MMD compares the mean embeddings in a feature space, facilitating its use as a loss function in various machine learning tasks, including density estimation, generative modeling, and inverse problems tackled with invertible neural networks. Its simplicity and robust theoretical foundations make MMD particularly advantageous.

To compute MMD for multi-modal data, a product measure is constructed to create a new probability space. Combining two probability spaces increases the complexity of the resulting σ-algebra, necessitating additional samples to characterize the probability space, as noted in Fact 3. We propose Equation 5 to define an induced measure, specifically the anchor-aligned probability measure, as a replacement for the traditional product measure in MMD computation. Equation 5 is essential for the application of MMD in anchor-aligned feature spaces. The transformation of the original probability measure P_x via an anchor-aligned mapping is demonstrated. This equation defines a new probability measure, $P_x^{a_y}$, corresponding to the anchor a_y.

Equation 6 specifies the MMD loss between two distributions, $\mathbb{P}_x^{id}$ (in-domain) and $\mathbb{P}_x^{ood}$ (out-of-domain), within the anchor-aligned feature space. This equation quantifies the discrepancy between two probability distributions in the anchor-aligned feature space. Here is how to use Equation 6 for the current task: The first term computes the expectation over all x_{id} samples, while the second term computes the expectation over all x_{ood} samples. Here, k is the kernel function, and in our experiments, we use the Gaussian kernel. f is the image encoder, and θ is being updated during training. In practical implementations, the expectations are replaced with sample averages.

C. Experiments Protocol

C.1. Base-to-Novel Dataset Split

In our experiment, we partition all class samples into two distinct groups, as outlined in the tables: the Base group (Table 1) and the Novel group (Table 2).

Consider the ImageNet dataset illustrated in Table 3, which consists of 1,000 classes. We divide the training set into two subsets, each containing 500 non-overlapping classes. For instance, one subset may include classes such as ["tench", "goldfish", "great white shark", "tiger shark", ...], and the other might feature ["spindle", "sports car", "spotlight", ...]. This separation ensures that no class from one group appears in the other, thereby preventing the model from encountering unknown classes during training and enhancing the fairness and credibility of our out-of-distribution evaluations. The test set follows a similar bifurcation, maintaining correspondence with the class labels from the training set.

	Classes	Train-Samples	Val-Samples	Test-Samples
OxfordPets	18	288	368	1881
Flowers102	51	816	817	1053
FGVCAircraft	50	800	1667	1666
DTD	23	368	564	864
EuroSAT	5	80	2700	4200
StanfordCars	98	1568	818	4002
Food101	50	800	10100	15300
SUN397	198	3168	1985	9950
Caltech101	50	800	825	1549
UCF101	50	800	949	1934
ImageNet	500	8000	25000	25000

Table 1. **Base class samples statistics.** The first column "Classes" denotes the total number of classes for each category. The columns "Train-Samples", "Val-Samples", and "Test-Samples" represent the respective number of images allocated for model training, validation, and testing purposes.

	Classes	Train-Samples	Val-Samples	Test-Samples
OxfordPets	19	304	368	1788
Flowers102	51	816	816	1410
FGVCAircraft	50	800	1,666	1667
DTD	24	384	564	828
EuroSAT	5	80	2,700	3900
StanfordCars	98	1568	817	4039
Food101	51	816	10,100	15000
SUN397	199	3184	1,985	9900
Caltech101	50	800	824	916
UCF101	51	816	949	1849
ImageNet	500	8000	25000	25000

Table 2. **Novel class samples statistics.** The first column "Classes" denotes the total number of classes for each category. The columns "Train-Samples", "Val-Samples", and "Test-Samples" represent the respective number of images allocated for model training, validation, and testing purposes.

	Classes	Train-Samples	Val-Samples	Test-Samples	Task
OxfordPets	37	2944	736	3669	Fine-Grained
Flowers102	102	4093	1633	2463	Fine-Grained
FGVCAircraft	100	3334	3333	3333	Fine-Grained
DTD	47	2820	1128	1692	Textures
EuroSAT	10	13500	5400	8100	Satellite Images
StanfordCars	196	6509	1635	8041	Fine-Grained
Food101	101	50500	20200	30300	Food
SUN397	397	15880	3970	19850	Scene
Caltech101	100	4128	1649	2465	Object
UCF101	101	7639	1898	3783	Action
ImageNet	1000	12800000	N/A	50000	Object

Table 3. **All class samples statistics from the original datasets.** The last column "task" provides a broad categorization of these image classification tasks, such as fine-grained classification or texture classification.

C.2. Group Robustness Baseline

For the group robustness experiment described in Section 4.4, we give a more comprehensive introduction about the baseline method that we compared with.

We evaluate our method against several methods in group robustness experiments, including zero-shot classification, ERM linear probing [**?**], and ERM adapter training [**?**]. Additionally, we compare against recent approaches tailored to enhancing downstream transfer in analogous scenarios, all while utilizing only pretrained model embeddings [**?**].

One such method is Weight space ensembling (WiSE-FT) [**?**], which initially trains a linear classifier using standard ERM and then combines the classifier outputs with the initial zero-shot predictions. Although originally proposed for training linear classifiers and fine-tuning the original weights of a foundational model, we focus on the prompt tuning in the extra parameter in our context.

Another approach is Deep feature reweighting (DFR) [**?**], which entails training a linear probe on embeddings computed from a pretrained model over group-balanced data. Similar to previous studies [**?**, **?**], we treat incorrectly and correctly classified samples as proxies for distinct groups.

Lastly, we consider the Contrastive Adapter approach [**?**], which introduces contrastive adapting. This method trains adapters with contrastive learning to bring sample embeddings closer to both their ground-truth class embeddings and other sample embeddings within the same class. While our method differs from this work, as we apply Contrastive learning to Prompt Tuning instead of Adapters.

C.3. Training Details

We utilized SGD as the optimizer optimizer with an initial learning rate of 0.0025 for Batch size 4, and a learning rate of 0.01 for batch size 128. The cosine annealing strategy is chosen to schedule the learning rate. For the Base to Novel Generalization setting, we use a few-shot training of 16 shots with a training duration of 20 epochs, while for Group Roubustness, we train 10 epochs on Waterbird and 5peochs on CelebA for the full dataset. All images were resized to 224x224 pixels, utilizing the same image preprocessing technique for the CLIP image encoder. All CLIP models adopted the ViT-B/16 backbone. We maintained consistency across all other settings as the baseline work, making modifications solely to the loss function to ensure a fair comparison between our method and the standard cross-entropy loss.

	Method	Pets	Flowers	Aircraft	DTD	EuroSAT	Cars	Food	Caltech	UCF	Avg
Base	G-Means	95.4	97.6	43.2	82.9	91.9	78.6	90.7	98.2	86.9	85.0
	H-Cluster	95.2	97.5	43.1	83.0	92.2	78.7	90.5	98.0	86.7	85.0
	K-Means	95.3	97.5	43.0	83.3	92.4	78.8	90.6	98.1	86.5	**85.1**
Novel	G-Means	97.5	77.2	38.2	63.6	68.1	75.4	91.7	94.5	78.2	76.0
	H-Cluster	97.4	77.1	36.7	64.1	72.5	74.9	90.5	94.2	78.5	76.2
	K-Means	97.5	77.7	36.9	63.9	79.4	75.2	91.7	94.1	79.1	**77.3**

Table 4. **Anchor Selection Method Comparison.** K-means is the default anchor selection method used in this paper. G-Means represents the group means anchor selection method. H-Cluster means hierarchical clustering anchor selection method. The 'Avg' represents the average accuracy over all the datasets.

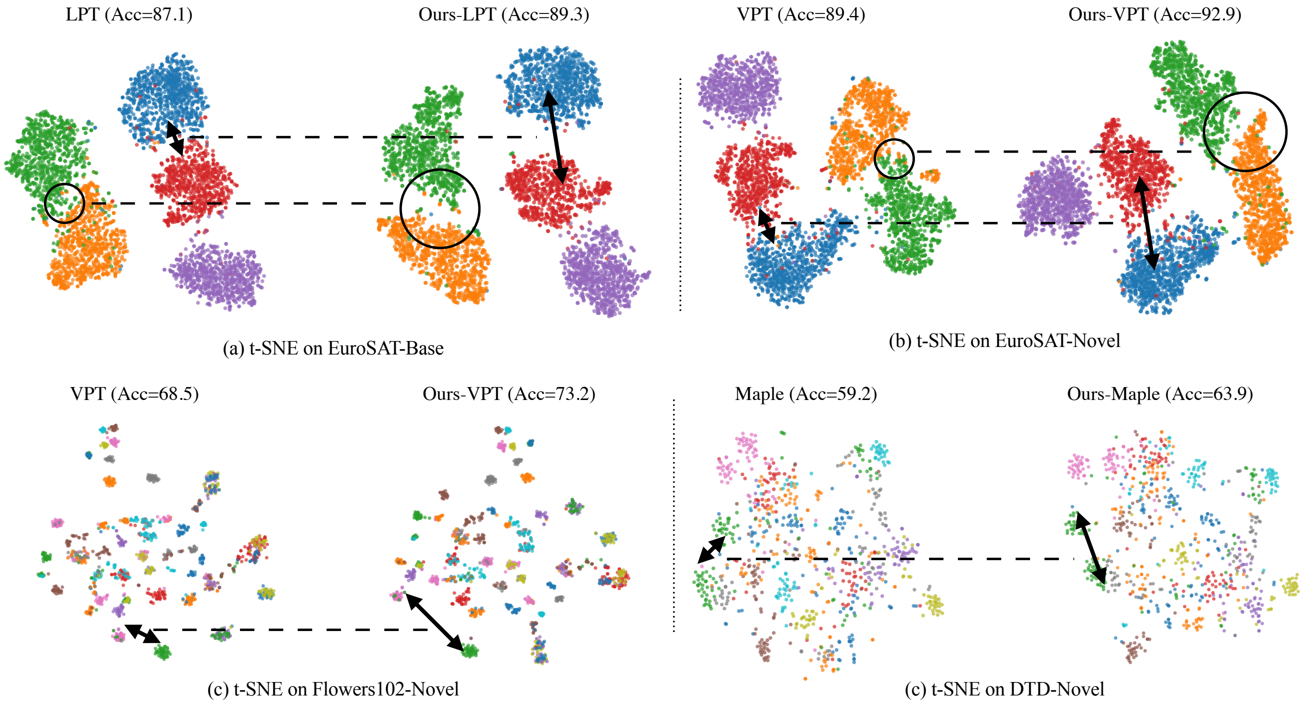
D. More Experiments Results

D.1. Anchor Selections Comparison

To evaluate whether different static anchor selections affect the final results, we conducted the ablation study on the anchor selection experiment, with the results shown in Table 4. We used the pre-trained CLIP model with a ViT-B/16 backbone as the feature extractor. All training images from each dataset were fed into the model's image encoder, and the resulting features were stored. The features are grouped by the ground truth label, then we use different anchor selection methods to choose the most representative one as the static anchor. The anchor selection methods we have are (1) K-means: we utilize the cluster center of K-means as the static anchor; (2) Hierarchical clustering: also the cluster center is utilized as the static anchor; (3) Group means, we direct calculation of the mean features for all the samples in each group. Table 4 shows that K-means and other methods do not have significant differences, while K-means yield better results compared to the hierarchical clustering method.

D.2. t-SNE Visualization

We show more t-SNE visualization results in Figure 1. In Figure (a), it is evident that applying our $\mathcal{L}_{\text{Aligned}}$ method to LPT increases the distance between cluster centers of the green color point and the orange color points. This indicates that our method enhances the learned latent space, bringing it closer to real samples, strengthening the model's decision



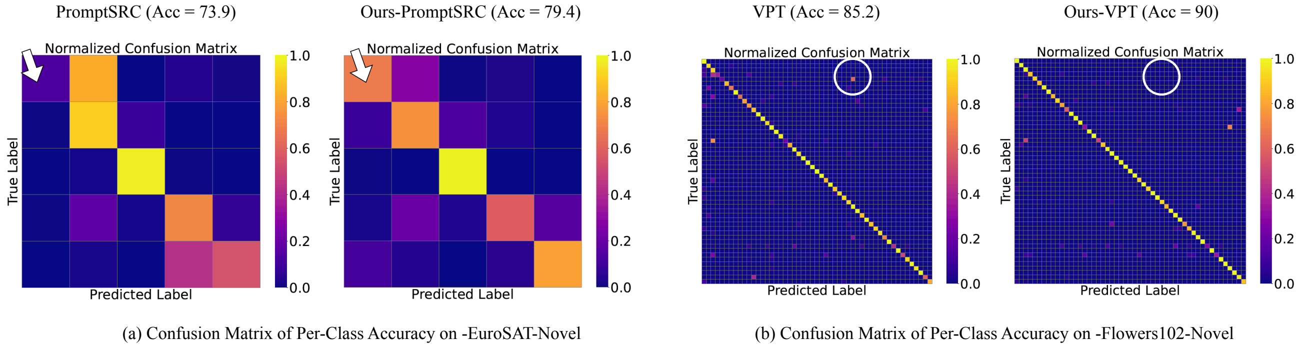
(a) t-SNE on EuroSAT-Base

(b) t-SNE on EuroSAT-Novel

(c) t-SNE on Flowers102-Novel

(c) t-SNE on DTD-Novel

Figure 1. **The t-SNE Visualization of Latent Embeddings.** The arrows in the figures illustrate our method can push the boundary between the two categories further apart. The circles in Figures (a) and (b) demonstrate that our method can separate the overlapping features of the two categories away from each other.



(a) Confusion Matrix of Per-Class Accuracy on -EuroSAT-Novel

(b) Confusion Matrix of Per-Class Accuracy on -Flowers102-Novel

Figure 2. **The Confusion Matrix for Per-Class Accuracy.** For Figure (a), without our method, the category in the first row is misclassified as the second category. After using our method, the first category classification is successfully made to achieve the highest accuracy.For Figure (b), our method also significantly improves one misclassified subclass, thereby improving the overall accuracy on the entire task.

boundaries, and consequently improving its accuracy. Similar improvements are observed in Figures 1 (b) (c) and (d). Additionally, the circle in Figure 1 (a) and Figure (b) shows that by using our method, we separate the overlap clusters to no-overlap clusters, which also confirms the effectiveness of our $\mathcal{L}_{\text{Aligned}}$ method.

D.3. Confusion Matrix Comparison

To conduct a more granular analysis of the performance improvements brought about by our method, we visualized the confusion matrices representing the accuracy for each category. The experimental results are illustrated in Figure 2. In Figure 2 (a), in the PromptSRC classification experiment on the EuroSAT dataset, the highest value in the first row of the baseline confusion matrix deviated from the diagonal, representing the Pasture Land category, with an accuracy of only 13.2%. Upon utilizing our $\mathcal{L}_{\text{Aligned}}$ loss function, the first row of the confusion matrix aligned with the diagonal, and the classification accuracy for Pasture Land improved to 68%, which lead to the all-class accuracy improved to 79.4%. Similarly in Figure 2 (b), the figure shows the classification experiments of VPT on the Oxford Flowers dataset. In the confusion matrix of the baseline model, we observed that the classification accuracy for the fifth category, English Marigold, deviated significantly from the diagonal, with an accuracy of only 20%. After applying our proposed $\mathcal{L}_{\text{Aligned}}$ loss function, the classification accuracy for English Marigold increased to 90%.

E. Limitations

Our method aims to construct relative representations in the latent space for cross-modal alignment between image and text modalities, utilizing both static and stochastic anchors. A significant limitation of this method is its high dependency on the selection of the Anchor. For instance, if the static anchor selected does not accurately capture the clustering characteristics of the targeted image category, it may result in biased cross-modal alignment, thereby adversely affecting the learning performance of the model. Additionally, in complex or non-standardized scenarios, finding a suitable static anchor point can be challenging, which constrains the general applicability of our approach

F. Broader Impact

Our proposed approach offers an effective technique applicable to visual language models characterized by an Image-Text dual-branch architecture, which is plug-and-play and can be integrated with many existing prompt tuning methods. Consequently, applying our method to the more sophisticated Prompt Tuning framework could yield further enhancements in performance. We leave these explorations for future research.