LEARNING ADAPTIVE AND VISUALLY ALIGNED CONDITIONS FOR GENERATIVE ZERO-SHOT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative zero-shot learning (ZSL) synthesizes visual features for unseen classes by learning a semantic-conditioned generator from seen classes. Since semantic conditions determine the knowledge transfer from seen to unseen classes, their effectiveness is critical for generating plausible features. However, existing class-level semantic prototypes lack semantic diversity to capture intra-class variations, while exhibiting inconsistent inter-class structures with the visual space due to the semantic–visual gap. To address these issues, we propose Adaptive Attribute Distribution and Visual Structure Alignment (AAVS), a unified semantic condition learning framework for generative ZSL. Specifically, the Adaptive Attribute Distribution (AAD) learns dimension-adaptive attribute distributions with discriminability-guided variation calibration to capture transferable semantic diversity, while Visual Structure Alignment (VSA) aligns the diverse attributes with refined visual prototypes to learn visual inter-class structures. By jointly modeling semantic diversity and visual structural consistency, AAVS provides more effective generation conditions for unseen classes to generate visual features. Extensive quantitative and qualitative evaluations on three benchmarks demonstrate that AAVS outperforms existing generative ZSL methods and validate the effectiveness of the learned generation conditions.

1 INTRODUCTION

Conventional visual recognition relies on a closed-set assumption: every test class must be represented in the training data, which is difficult to satisfy in real-world applications, where new classes may emerge without labeled data. Zero-shot learning (ZSL) (Lampert et al., 2009; Xian et al., 2019a) offers a promising solution by leveraging semantic information to transfer knowledge learned from seen classes to unseen classes, enabling recognition without visual examples. With the development of generative models, including variational autoencoders (VAEs) (Kingma & Welling, 2014), generative adversarial networks (GANs) (Goodfellow et al., 2014), and diffusion models (Ho et al., 2020; Xiao et al., 2022), generative zero-shot learning (Generative ZSL) (Xian et al., 2018) further learns a semantic-conditioned generator from seen classes and uses it to synthesize visual features for unseen classes, which has consequently become an effective paradigm to ZSL.

Since the generator can only be trained on seen classes, the semantic conditions play a crucial role in transferring its generation capability from seen to unseen classes to synthesize plausible features for unseen classes. Existing methods typically adopt class-level semantic conditions, such as predefined attributes (Li et al., 2023; Chen et al., 2023b; Hou et al., 2024; Chen et al., 2025b; Hou et al., 2026) or text embeddings (Reed et al., 2016; Wang et al., 2023; Chen et al., 2025a), which build an effective bridge for knowledge transfer. however, two inherent yet usually overlooked Challenges remain:

1) Static semantic conditions fail to accurately describe intra-class instance variations. As shown in Fig. 1(a), the class-level attributes of Collie assign a high value to **Black** and a low value to **Orange**, which well describes the black-and-white Border Collie but fails to represent the warm-orange Rough Collie. Consequently, instances within the same class share identical semantic conditions, leaving the generator without instance-varying semantic guidance to capture intra-class variations. This limits the learning of accurate visual–semantic correspondence (Ye et al., 2025) and results in insufficient diversity of generated features. Recent methods attempt to alleviate this issue by leveraging visual features from seen classes to obtain instance-level semantic enhancements dur-

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.

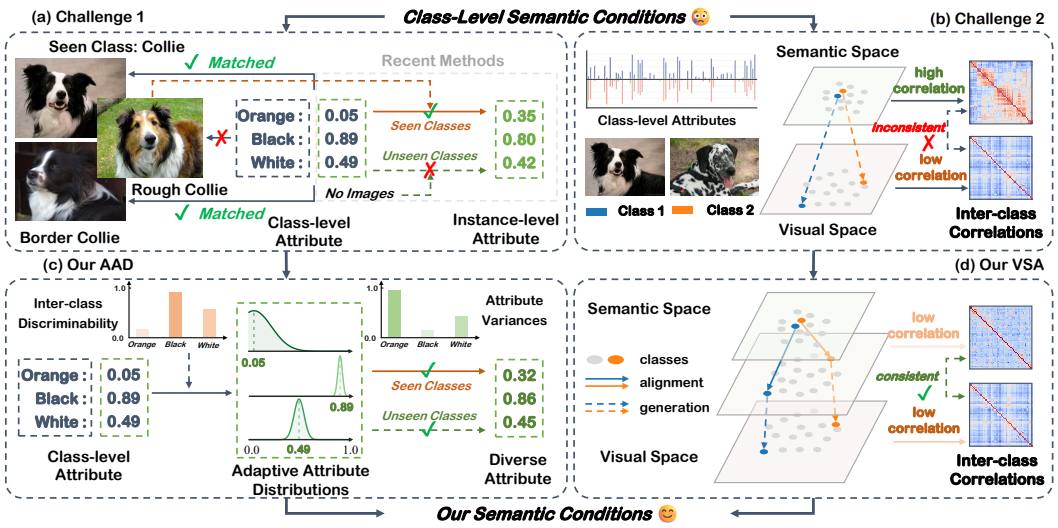

Figure 1: Challenges of class-level semantic conditions and our motivation. (a) Challenge 1 ✗: Class-level semantics lack diversity to describe intra-class variations. Although recent methods introduce visual-guided semantic enhancement, they rely on seen visual features and are hard to generalize to unseen classes . (b) Challenge 2 ✗: Semantically similar classes may exhibit distinct visual appearances (left), revealing the semantic-visual gap and inconsistent inter-class structures between two spaces (right). (c) Our AAD ✓: Different attributes contribute differently to class discrimination, motivating attribute-adaptive distribution learning to obtain diverse semantic conditions without visual features. (d) Our VSA ✓: Semantic representations are aligned with visual structures to obtain visually consistent generation conditions.

ing generator training (Chen et al., 2023b; Hou et al., 2024; Jiang et al., 2024). However, since visual features of unseen classes are unavailable, such visual-guided semantic enhancements are hard to obtain diverse semantics for unseen classes and still rely on a single semantic condition.

2) Class-level semantic conditions exhibit inconsistent inter-class structures with the real visual space. As shown in Fig. 1(b), two classes with nearly identical attributes but exhibit markedly different visual appearances. Such mismatch reflects the semantic–visual gap Chen et al. (2023b; 2025b), which further leads to discrepancies in inter-class structures between two domains, as illustrated by the correlation matrices in Fig. 1(c). Since generative ZSL relies on semantic conditions to capture inter-class structures and learn cross-domain mappings, this structural inconsistency makes knowledge transfer from seen to unseen classes more challenging, causing generated features to deviate from the real visual distribution.

These observations raise a fundamental question: **what properties should an effective semantic condition possess for generative ZSL?** We answer this question from two perspectives:

1) Semantic diversity. Class-level semantic conditions fail to describe intra-class instance variations due to their limited semantic diversity. Since visual features within the same class exhibit rich variations as shown in Fig. 1(a), semantic conditions should also provide a variation space to characterize potential instance-level semantics. Furthermore, different attributes contribute differently to class discrimination. As shown in Fig. 1(c), **Black** and **White** is more discriminative for Collie than **Orange**, indicating that different attributes play different roles in representing class semantics and variations. Motivated by this observation, we propose **Adaptive Attribute Distribution (AAD)**, which learns adaptive attribute distributions for different attribute dimensions and adjusts their variation ranges according to attribute discriminability. By learning entirely from the shared semantic space, AAD can naturally transfer to unseen classes and provide diverse semantic conditions for feature synthesis.

2) Structural consistency. Due to the semantic-visual gap, the inter-class structures encoded by class-level semantic conditions are inconsistent with the visual space, as shown in Fig. 1(b). Since generators tend to inherit the structures from conditioning spaces, such discrepancies can be prop-

agated to generated features, causing them to deviate from the real visual distribution. To obtain semantic conditions with visual structural consistency, we propose **Visual Structure Alignment (VSA)**, which projects attributes into the visual space and aligns the embeddings with the refined visual prototypes to learn visual inter-class structures. As shown in Fig. 1(d), the aligned attribute embeddings exhibit more consistent inter-class structures with the visual space, thereby enabling more reliable cross-domain feature generation and mitigating the distribution shift between generated and real features.

To this end, we propose **AAVS**, a unified semantic condition learning framework consisting of Adaptive Attribute Distribution (AAD) and Visual Structure Alignment (VSA). Starting from class-level semantic conditions, AAVS learns semantic representations with both semantic diversity and visual structural consistency. These representations jointly serve as generation conditions, enabling the generator to establish more accurate visual-semantic correspondence. Since AAVS is learned solely from the shared attribute space, it can be effectively transferred to unseen classes and generate plausible visual features that better match the real visual distribution.

Our contributions are summarized as follows:

- To address the limitation that class-level semantic conditions cannot describe instance variations, we propose **Adaptive Attribute Distribution**, which constructs adaptive attribute distributions to obtain semantically diverse attributes.
- To address the class-level semantic conditions' inconsistent inter-class structures with the real visual space, we propose **Visual Structure Alignment**, which projects attributes into the visual space to obtain attribute embeddings with visual structural consistency.
- Extensive experiments on widely used benchmarks demonstrate that our method outperforms state-of-the-arts and can obtain effective generation conditions.

2 Related Work

2.1 Generative Zero-Shot Learning

Generative Zero-shot Learning synthesizes visual features for unseen classes by using conditional generative models, such as GANs (Goodfellow et al., 2014), VAEs (Kingma & Welling, 2014) and Diffusion Models (Ho et al., 2020). Focusing on improving generated feature quality, existing methods are mainly divided into two lines: (1) Visual feature enhancement. Some approaches (Narayan et al., 2020; Chen et al., 2021; 2025b; Ye et al., 2025) improve visual feature by fine-tuning the backbone or adapting pretrained visual features, while other approaches (Kong et al., 2022; Li et al., 2023) further exploit visual-semantic alignment to obtain semantically guided visual representations. (2) Semantic condition enhancement. Early methods explore richer semantic representations, such as text embeddings Reed et al. (2016) and CLIP-based representations Wang et al. (2023); Chen et al. (2025a), whereas recent approaches attempt to evolve semantic prototypes under the guidance of visual information (Chen et al., 2023b; Hou et al., 2024; Ye et al., 2025). Despite the progress, existing semantic enhancement relies on visual information, making it challenging to transfer to unseen classes where visual samples are unavailable. In contrast, our Adaptive Attribute Distribution directly learns transferable attribute variation patterns and obtains diverse semantic conditions for unseen classes.

2.2 Embedding-based Zero-Shot Learning

Embedding-based zero-shot learning represents another mainstream paradigm for ZSL, where the semantic-visual gap remains a central challenge. Early methods (Zhang et al., 2017; Xian et al., 2016) learn a shared embedding space to align visual features with semantic representations, and recent methods (Chen et al., 2023a; 2024; 2025c) further leverage attention mechanisms to capture discriminative regions. In addition, semantic-guided methods (Jiang et al., 2025; Liu et al., 2025a;b; Zhang et al., 2026) exploit attribute prototypes to enhance semantic-aware representation learning. In generative settings, semantic information directly controls visual feature synthesis, and the semantic-visual gap may cause the generated features to inherit incorrect inter-class structures. To address this issue, our Visual Structure Alignment incorporates visual structures into generation conditions before synthesis, enabling conditions that better reflect visual relationships.

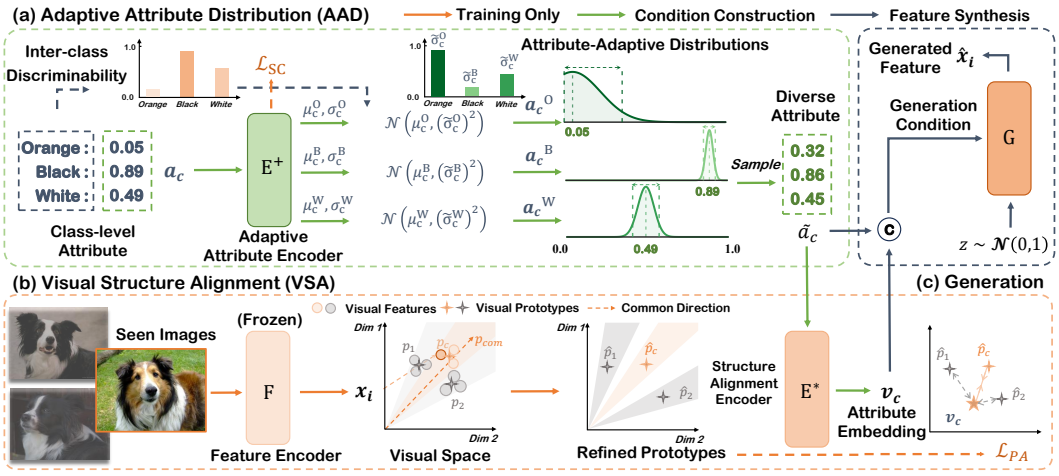

Figure 2: Overview of AAVS. Given class attributes, AAVS constructs generation conditions via →: (a) Adaptive Attribute Distribution (AAD), learns adaptive attribute distributions with inter-class discriminability calibrating the variation magnitude and semantic consistency constraint $\mathcal{L}_{SC}$, producing diverse attributes $\tilde{a}_c$. And (b) Visual Structure Alignment (VSA), aggregates seen features into prototypes and refines them by removing dominant common directions (→), which supervise the attribute embedding v_c through prototype alignment loss $\mathcal{L}_{PA}$ to incorporate visual class structures. In the (c) generation phase (→), the resulting a_c^+ and v_c are concatenated as generation conditions, with latent noise z to synthesize visual features.

2.3 Effective Generation Condition

Generative models learn to synthesize samples that follow the real data distribution (Goodfellow et al., 2014). VAEs and GAN-based models introduce latent noise sampled from Gaussian priors to capture stochastic variations (Kingma & Welling, 2014; Goodfellow et al., 2014; Bau et al., 2019; Larsen et al., 2016). However, latent noise alone provide limited guidance for modeling complex high-dimensional distributions, resulting in incomplete mode coverage and limited diversity (Metz et al., 2017; Arjovsky et al., 2017; Bau et al., 2019). To enhance controllability, high-level semantic information is incorporated as generation conditions to provide semantic constraints and capture inter-class relationships (Mirza & Osindero, 2014; Xian et al., 2018). Nevertheless, semantic-visual structural inconsistency may cause generated samples to deviate from the real visual distribution (Chen et al., 2023b; Hou et al., 2024; Chen et al., 2025b). These observations emphasize the need for generation conditions that captures intra-class variations while preserving visual-consistent inter-class structures, motivating us to propose AAVS, which learns adaptive attribute distributions and aligns semantic embeddings with visual structures to construct more effective generation conditions.

3 Methodology

3.1 Notations

Generative ZSL learns a semantic-conditioned generator from seen classes to synthesize unseen-class visual features. The seen and unseen datasets are defined as $\mathcal{D}^s = \{(x_i^s, y_i^s)\}_{i=1}^{N^s}$ and $\mathcal{D}^u = \{(x_i^u, y_i^u)\}_{i=1}^{N^u}$, where $\mathcal{Y}^s \cap \mathcal{Y}^u = \emptyset$ and unseen visual features are unavailable during training. Following Xian et al. (2019a), seen datsets are split into training and test sets. Each class $c \in \mathcal{Y}^s \cup \mathcal{Y}^u$ is associated with a semantic attribute $a_c \in \mathbb{R}^{|A|}$. Conventional ZSL (CZSL) learns $f_{CZSL} : \mathcal{X} \to \mathcal{Y}^u$, while generalized ZSL (GZSL) learns $f_{GZSL} : \mathcal{X} \to \mathcal{Y}^s \cup \mathcal{Y}^u$.

3.2 Adaptive Attribute Distribution

Motivation. As illustrated in Fig. 1(a), static class-level semantic conditions cannot capture intra-class variations, leading to inaccurate visual–semantic correspondence and limited diversity of gen-

erated features. Although visual-guided semantic enhancement introduces instance-level semantics during training, it's difficult to be transferred to unseen classes due to the absence of visual features Chen et al. (2023b); Hou et al. (2024). Meanwhile, different attributes contribute differently to class discrimination (Fig. 2(a)), motivating us to learn attribute-adaptive variation patterns. Therefore, we propose **Adaptive Attribute Distribution (AAD)**.

Adaptive Attribute Distribution. Since intra-class visual variations usually occur around a class prototype, we aim to learn semantic variation patterns around the semantic prototype. As shown in Fig. 2(a), given a class-level attribute a_c, AAD learns latent attribute variation patterns for each dimension. Specifically, we introduce an Adaptive Attribute Encoder E^{+}, which predicts Gaussian distribution parameters from $a_c = [a_c^1, \cdots, a_c^{|A|}] \in \mathbb{R}^{|A|}$:

$$\mathrm{E}^{+}(a_c) = \{(\mu_c^j, \sigma_c^j)\}_{j=1}^{|A|}, \tag{1}$$

where μ_c^j and σ_c^j denote the mean and standard deviation of the j-th attribute dimension. Then the attribute variation is sampled as: $\Delta a_c^j \sim \mathcal{N}(\mu_c^j, (\sigma_c^j)^2)$, where Δa_c^j represents the variation in the j-th attribute dimension. By using the a_c as a semantic anchor, we obtain the diverse attribute:

$$\tilde{a}_c^j = a_c^j + \lambda \cdot \Delta a_c^j, \tag{2}$$

where λ controls the magnitude of the learned variation. Since instance-level semantic supervision is unavailable, uncontrollable variations may introduce unstable distributions during training. Therefore, we introduce a semantic consistency constraint $\mathcal{L}_{SC}$ to regularize the learned distributions while preserving semantic consistency:

$$\mathcal{L}_{SC} = D_{KL}(\mathcal{N}(\mu_c^j, (\sigma_c^j)^2) \| \mathcal{N}(0, 1)).$$

Discriminability-guided Variation Calibration. Learned Attribute variations may weaken discriminative information. To balance diversity and discrimination, AAD calibrates the variation range according to attribute discriminability. We assume that highly discriminative attributes should vary less, while less discriminative attributes can accommodate larger variations. For the j-th attribute dimension, its discriminability is measured as:

$$d_j = \frac{1}{|\mathcal{Y}^s|(|\mathcal{Y}^s| - 1)} \sum_{c \neq k} (a_c^j - a_k^j)^2. \tag{3}$$

A larger d_j indicates stronger class-discriminative ability. After normalization, the standard deviation is calibrated as:

$$\tilde{\sigma}_c^j = (1 - \bar{d}_j)\sigma_c^j, \tag{4}$$

where $\bar{d}_j$ denotes the normalized discriminability. With the calibrated standard deviation, the final attribute variation is sampled as: $\Delta \tilde{a}_c^j \sim \mathcal{N}(\mu_c^j, (\tilde{\sigma}_c^j)^2)$, and the final diverse attribute is obtained as $\tilde{a}_c^j = a_c^j + \lambda \cdot \Delta \tilde{a}_c^j$. Since attribute variations capture class-shared variation patterns, they can be transferred to unseen classes.

3.3 Visual Structure Alignment

As illustrated in Fig. 1(b), structurally misaligned semantic conditions may cause the synthesized features to deviate from the real visual distribution. To address this issue, we propose **Visual Structure Alignment (VSA)** to learn semantic representations with visual-consistent inter-class structures, as shown in Fig. 2(b). During training, we use a frozen feature encoder F to extract visual features from seen-class images. Since pre-trained visual representations may contain class-shared common directions, we first compute class prototypes: $p_c = \frac{1}{N_c} \sum_{i:y_i=c} x_i$, and then suppress the dominant common direction identified by SVD:

$$\hat{p}_c = p_c - \alpha(p_c^\top p_{\mathrm{com}}) p_{\mathrm{com}}, \tag{5}$$

where p_{com} is the unit-normalized leading right singular vector of the seen-class prototype matrix and α controls the suppression strength. The refined prototypes enhance inter-class discriminability while preserving class structures. Based on $\tilde{a}_c$ learned by AAD, we introduce a Structure Alignment Encoder E^{*} to obtain attribute embeddings: $v_c = \mathrm{E}^{*}(\tilde{a}_c)$, optimized by the prototype alignment loss $\mathcal{L}_{PA}$:

$$\mathcal{L}_{PA} = -\log \frac{\exp(v_c^\top \hat{p}_c / \tau)}{\sum_{k \in \mathcal{Y}^s} \exp(v_c^\top \hat{p}_k / \tau)}. \tag{6}$$

This prototype-guided alignment enables v_c to capture discriminative visual structures while maintaining the semantic transferability of attribute-based conditions.

Table 1: Comparison with state-of-the-art methods on AWA2, CUB, and SUN. Acc, U, S, and H (%) are reported. Avg. H averages H across all three datasets when available. † indicates methods using class names as semantic conditions. Best and second-best results are shown in **bold** and underlined, respectively and – denotes unreported results.

Method	Venue	AWA2				CUB				SUN				Avg. H
		Acc	U	S	H	Acc	U	S	H	Acc	U	S	H	
Embedding-based														
CLIP†	ICML'21	–	–	–	–	–	55.2	54.8	55.0	–	–	–	–	–
PSVMA	CVPR'23	–	73.6	77.3	75.4	–	70.1	77.8	73.8	–	61.7	45.3	52.3	67.2
ZSLViT	CVPR'24	70.2	66.1	84.6	74.2	78.9	69.4	78.2	73.6	68.3	45.9	48.3	47.3	65.0
ZeroMamba	AAAI'25	71.9	67.9	87.6	76.5	80.0	72.1	76.4	74.2	72.4	56.5	41.4	47.7	66.1
CSAP-ZSL	AAAI'25	–	66.8	84.4	74.6	–	68.1	73.2	70.6	–	61.3	45.5	52.2	65.8
AENet	AAAI'25	75.2	70.3	80.1	74.9	80.3	73.1	76.4	74.7	70.4	58.6	45.2	51.0	66.9
VSPCN	CVPR'25	76.6	71.8	84.3	77.6	80.6	72.8	78.9	75.7	75.3	59.4	49.1	53.8	69.0
SVIP	ICCV'25	69.8	65.4	87.7	74.9	79.8	72.1	78.1	75.0	71.6	53.7	48.0	50.7	66.9
PSVMA+	TPAMI'25	79.2	74.2	86.4	79.8	78.8	71.8	77.8	74.6	74.5	61.5	49.4	54.8	69.7
SAGE	AAAI'26	83.7	78.1	87.2	82.4	82.1	**75.1**	77.5	76.3	**77.2**	63.7	52.2	**57.4**	72.0
Generative-based														
f-VAEGAN	CVPR'19	71.1	57.6	70.6	63.5	61.0	48.4	60.1	53.6	64.7	45.1	38.0	41.3	52.8
TF-VAEGAN	ECCV'20	72.2	59.8	75.1	66.6	64.9	52.8	64.7	58.1	66.0	45.6	40.7	43.0	55.9
ICCE	CVPR'22	78.4	65.3	82.3	72.8	72.7	67.3	65.5	66.4	–	–	–	–	–
ZeroNAS	TPAMI'22	73.2	61.4	75.3	67.6	66.4	56.0	63.8	59.6	68.3	47.1	41.8	44.3	57.2
CDL + OSCO	TPAMI'23	–	48.0	71.0	57.1	–	–	–	–	–	32.0	**65.0**	42.9	–
DSP	ICML'23	71.6	60.0	86.0	70.7	62.8	51.4	63.8	56.9	68.6	48.3	43.0	45.5	57.7
TF-VAEGAN + SHIP†	ICCV'23	–	61.2	**95.9**	74.7	–	22.5	**82.2**	35.3	–	–	–	–	–
DML	ACCV'24	–	62.2	82.3	70.9	–	57.1	81.6	67.2	–	39.6	52.7	45.9	61.3
VADS	CVPR'24	82.5	75.4	83.6	79.3	**86.8**	74.1	74.6	74.3	76.3	**64.6**	49.0	55.7	69.8
ViFR	IJCV'25	77.8	68.2	78.9	73.2	74.5	63.9	72.0	67.7	69.2	51.3	40.0	44.7	61.9
AAVS (Ours)	–	**85.4**	**82.1**	85.0	**83.5**	79.2	74.2	81.0	**77.4**	75.8	58.9	54.0	56.3	**72.4**

3.4 Optimization and Unseen-class Feature Generation

Objective Loss Function. Overall, the total loss function is defined as follows:

$$\mathcal{L}_{total} = \mathcal{L}_G + \lambda_{SC}\mathcal{L}_{SC} + \lambda_{PA}\mathcal{L}_{PA}, \tag{7}$$

where $\mathcal{L}_G$ is the loss of the conditional generator G. λ_{SC} and λ_{PA} are the weights for $\mathcal{L}_{SC}$ and $\mathcal{L}_{PA}$. After training, we use the learned generator G to synthesize unseen-class visual features for ZSL classification. We also provide a detailed training algorithm in Appendix A.5.

Unseen Class Feature Generation. As shown by the green arrows in Fig. 2, given the class-level attribute a_u of an unseen class u, the trained E^+ first constructs adaptive attribute distributions through AAD, i.e., $\tilde{a}_u^j \sim \mathcal{N}(a_u^j + \lambda\mu_u^j, (\lambda\tilde{\sigma}_u^j)^2)$. The sampled attributes $\tilde{a}_u$ are then projected by E^* to obtain attribute embedding, i.e., $v_u = \mathrm{E}^*(\tilde{a}_u)$. Finally, as shown in Fig. 2(c), $\tilde{a}_u$ and v_u are concatenated as the generation condition, which is combined with sampled latent noise z to synthesize unseen-class visual features, i.e., $\hat{x}_u = \mathrm{G}(z; [\tilde{a}_u, v_u])$. Detailed evaluation procedures are in Appendix A.5.

4 Experiments

4.1 Datasets and Evaluation Protocols

We evaluate AAVS on three ZSL benchmarks: AWA2 (Xian et al., 2019a), CUB (Wah et al., 2011), and SUN (Patterson & Hays, 2012). The statistics of these datasets, including the numbers of images, classes, and attributes, are $\{37322, 50, 85\}$, $\{11788, 200, 312\}$, and $\{14340, 717, 102\}$, respectively. We follow the standard class split protocol (Xian et al., 2019a). Details are in Appendix A.4. For evaluation, we adopt the unified protocol (Xian et al., 2019a). We report unseen-class accuracy (Acc) in CZSL, and unseen (U), seen (S), and harmonic mean $\mathrm{H} = \frac{2\times\mathrm{S}\times\mathrm{U}}{\mathrm{S}+\mathrm{U}}$ in GZSL.

4.2 Implementation Details

We adopt f-VAEGAN (Xian et al., 2019b) as the conditional generator to synthesize visual features, which combines VAE and GAN for semantic-conditioned generation. Visual features are extracted from a pre-trained ViT (Oquab et al., 2023) using the CLS token. For the fine-grained CUB dataset, we further fine-tune ViT to alleviate dataset bias (Chen et al., 2021; 2025b). For the semantic consistency constraint, we set its weight λ_{SC} to 0.01 for all datasets to balance distribution learning and semantic consistency. The magnitude of learned variation λ (Eq. 2), the strength of common direction suppression are controlled by the parameter α (Eq. 5) and the number of synthesized features per class for AWA2, CUB, and SUN are $\{0.05, 0.1, 0.01\}$, $\{0.5, 0.5, 0.5\}$, and $\{1800, 500, 400\}$, respectively. All experiments are implemented in PyTorch and conducted on a single NVIDIA RTX 3090 GPU (24 GB).

4.3 Comparison with State-of-the-Art Methods

As shown in Table 1, we compare AAVS, built upon f-VAEGAN (Xian et al., 2019b), with state-of-the-art embedding-based and generative ZSL methods. Under CZSL, AAVS achieves 85.4% accuracy on AWA2, outperforming SAGE (Zhang et al., 2026) (83.7%) by 1.7%. It also obtains 79.2% and 75.8% on CUB and SUN, respectively, demonstrating competitive performance among generative methods. Under GZSL, AAVS achieves the highest average H of 72.4% among all compared embedding-based and generative methods. This verifies that AAD and VSA provide effective semantic conditions by capturing attribute variations and visual-consistent structures. Furthermore, AAVS remains competitive with recent ViT-based methods, including SVIP (Chen et al., 2025c), PSVMA+ (Liu et al., 2025b), and SAGE (Zhang et al., 2026). It also surpasses DSP (Chen et al., 2023b) and VADS (Hou et al., 2024), which rely on visual-guided semantic enhancement, highlighting the advantage of learning transferable attribute distributions without relying on visual features. VLM-based methods such as SHIP (Wang et al., 2023) achieve strong seen-class performance but exhibit limited generalization to unseen classes.

4.4 Ablation Study

To evaluate the contribution of each component in AAVS, we remove Adaptive Attribute Distribution (AAD), Visual Structure Alignment (VSA), and their key designs on AWA2, CUB, and SUN, as shown in Table 2. **(a)** Removing both AAD and VSA reduces AAVS to a baseline that directly uses class-level attributes for generation. Compared with the baseline, AAVS consistently improves performance across all datasets, e.g., increasing H from 65.4% to 83.5% on AWA2 and by 5.1% on CUB. This demonstrates that jointly modeling adaptive attribute distributions and visual structure alignment provides more effective generation conditions than static class-level semantics. **(b)** Removing AAD degrades performance by 11.0%, 4.3%, and 0.5% in H on AWA2, CUB, and SUN, respectively. This is because the model relies on static class-level semantics and lacks the ability to capture intra-class variations, highlighting the importance of semantic diversity. **(c)** Removing VSA leads to H decreases of 8.2%, 1.0%, and 2.5%, validating the importance of visual structure consistency for generating features aligned with real distributions. **(d, e)** Furthermore, removing either Eq. (4) or Eq. (5) consistently reduces performance, validating that variation calibration balances semantic diversity and class discrimination, while prototype refinement provides more discriminative supervision for visual structure alignment.

4.5 Qualitative Evaluation

4.5.1 Analysis of Learned Generation Conditions

Semantic Interpretability of Adaptive Attribute Distributions. To verify whether the learned attribute Distributions capture meaningful semantic patterns, we analyze their relationship with attribute inter-class discriminability (Eq. 3) on AWA2. We group attributes by semantic types (e.g., Black and White as global color) and compute the average variance and discriminability within each group, as shown in Fig. 3(a). The results reveal a negative correlation between discriminability and learned variation magnitude: highly discriminative attributes (e.g., size) exhibit smaller variations, while less discriminative attributes (e.g., pattern) allow larger variations, consistent with our moti-

Table 2: Ablation results of AAVS on AWA2, CUB, and SUN. AAVS w/o Eq. (4) ablates discriminability-guided variation calibration, while AAVS w/o Eq. (5) ablates visual prototype refinement. Avg. H averages H across all three datasets.

Configuration	AWA2				CUB				SUN				Avg. H
	Acc	U	S	H	Acc	U	S	H	Acc	U	S	H	
a) AAVS w/o AAD & VSA	64.1	62.4	68.7	65.4	77.1	66.8	78.7	72.3	73.7	57.4	50.0	53.5	63.7
b) AAVS w/o AAD	71.2	63.6	84.4	72.5	76.5	68.0	78.9	73.1	74.2	**59.9**	52.1	55.8	67.1
c) AAVS w/o VSA	78.8	70.6	80.6	75.3	**79.2**	73.2	79.9	76.4	73.9	57.4	50.6	53.8	68.5
d) AAVS w/o Eq. (4)	84.8	78.8	85.6	81.9	78.6	72.7	79.2	76.0	74.2	58.8	53.2	55.9	71.3
e) AAVS w/o Eq. (5)	85.2	78.2	**86.6**	82.3	78.9	74.1	79.8	76.8	75.3	59.7	52.3	55.8	71.6
f) AAVS	**85.4**	**82.1**	85.0	**83.5**	**79.2**	**74.2**	**81.0**	**77.4**	**75.8**	58.9	**54.0**	**56.3**	**72.4**

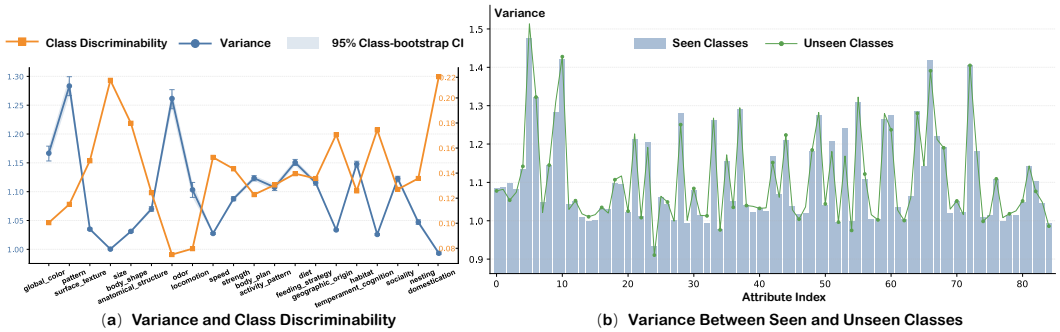

Figure 3: Qualitative analysis of the Adaptive Attribute Distribution (AAD) on AWA2. (a) Attribute discriminability versus learned variance σ: more discriminative attributes receive smaller variations, while less discriminative attributes allow larger variations. (b) Seen–unseen comparison of attribute-wise variation patterns, demonstrating that AAD transfers through the shared attribute space without unseen-class visual features.

vation. This indicates that AAD adaptively allocates variation capacity across attributes, preserving discriminative semantics while introducing additional diversity beyond the class prototype.

Transferability of Learned Attribute Variations. For generative ZSL, transferring diverse semantic conditions to unseen classes is essential. Fig. 3(b) compares attribute variation patterns between seen and unseen classes on AWA2. Although the variation patterns are learned only from seen classes, they exhibit similar attribute-level trends when applied to unseen classes. This demonstrates that AAD can transfer learned variation patterns through the shared attribute space, enabling unseen classes to obtain class-adaptive diverse attributes without visual observations or instance-level semantic annotations.

Visual Structure Consistency of Semantic Embeddings. Fig. 4 evaluates the effectiveness of VSA on the CUB dataset from two perspectives. As shown in Fig. 4(a), generated features largely inherit the inter-class structure of class attributes (box 1, Spearman's ρ of 0.73), validating our motivation. In contrast, VSA-learned attribute embeddings v exhibit stronger structural consistency with real visual features (box 2 and box 3, 0.45 vs. 0.51), indicating more visually aligned semantic conditions. Prototype refinement in Eq. (5) further alleviates similarity compression in pre-trained visual representations (details in Appendix A.1). After refinement, the inherited structural discrepancy is reduced from 0.73 to 0.31 (box 1 and box 1), while the alignment between attribute embeddings v and visual structures is improved (box 3, 0.57). The correlation matrices in Fig. 4(b) further visualize this enhanced alignment, demonstrating the effectiveness of VSA.

4.5.2 Quality of Generated Features

To qualitatively assess the generated features, we use the t-SNE (Van der Maaten & Hinton, 2008) to visualize them produced by f-VAEGAN (Xian et al., 2019b), f-VAEGAN with AAD, f-VAEGAN

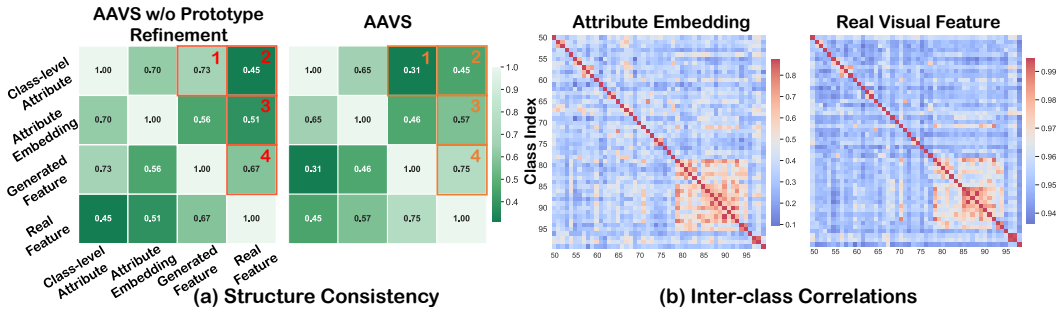

Figure 4: Qualitative evaluation of VSA on CUB. (a) Spearman's ρ (Spearman, 1961) matrices for four representations with and without prototype refinement; refinement increases the correlation between attribute embeddings and real visual features from 0.51 to 0.57 compared to class-level attributes (box 3 to box 3). (b) Inter-class correlation matrix of the attribute embeddings and real visual features, which also shows the effectiveness of VSA.

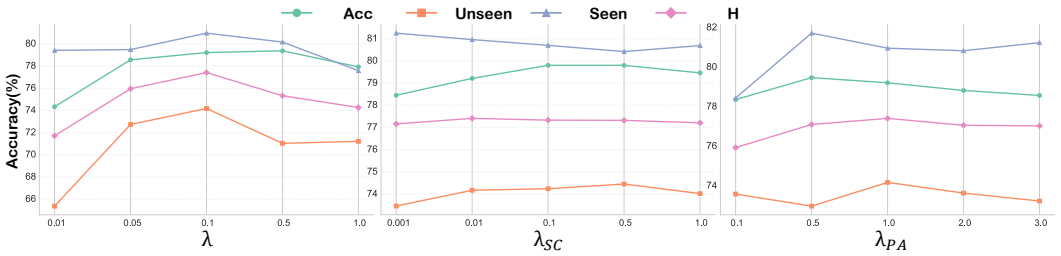

Figure 5: Hyper-parameter analysis on CUB. λ denotes the magnitude of learned variation in Eq. (2), λ_{SC} and λ_{PA} control the weights of two losses in Eq. (7), respectively.

with VSA, and the full AAVS model in Fig. 8. As detailed in Appendix A.3, the baseline generates overly compact intra-class distributions and exhibits a mismatch with the real inter-class structure. AAD alleviates the former by introducing adaptive attribute variations, while VSA addresses the latter through visually aligned attribute embeddings, as further illustrated by box 4 in Fig. 4(a). Combining both components, AAVS produces features with greater intra-class diversity and better consistency with the real visual structure, in line with our motivation.

4.6 Hyper-Parameter Analysis

Fig. 5 analyzes the sensitivity of λ, λ_{SC}, and λ_{PA} on CUB. As λ controls the magnitude of semantic variations in AAD, it determines the trade-off between diversity and consistency. Too small variations limit semantic diversity, while excessive variations deviate from semantic prototypes and harm performance, demonstrating the importance of proper variation control. In contrast, λ_{SC} and λ_{PA} remain stable across different values. Results on AWA2 and SUN are reported in Appendix A.2.

5 Conclusion

In this paper, we analyse the limitations of class-level semantic conditions in generative ZSL and identify two essential properties of effective generation conditions: semantic diversity and visual-structure consistency. Based on these properties, we proposed AAVS, in which AAD learns transferable, adaptive attribute distributions to produce diverse attributes, while VSA aligns attribute embeddings with visual class space to learn inter-class structure. Together, they construct effective semantic conditions that transfer from seen to unseen classes and guide the synthesis of higher-quality visual features. Extensive experiments on three benchmarks demonstrate that AAVS outperforms existing methods and validate the effectiveness of learned semantic conditions.

AI USE STATEMENT

In this work, we used generative AI tools for language editing and drafting portions of the manuscript. Generative AI was not used to generate or modify experimental data, conduct experiments, or create any figures or tables in the manuscript. All AI-assisted content was reviewed and revised by the authors. We take full responsibility for the final content of this work, including all text, claims, and materials produced with the assistance of generative AI.

ETHICS STATEMENT

This study uses publicly available ZSL benchmark datasets and does not involve human participants, private information, or newly collected data.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide a code repository in the supplementary material and dataset splits in the appendix, while all hyperparameters are specified in the main text. The repository includes features extraction scripts for all benchmark datasets, complete training and evaluation code, and detailed instructions for running the main experiments. All reported experiments can be conducted on a single NVIDIA RTX 3090 GPU.

REFERENCES

Martín Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *Proceedings of the International Conference on Machine Learning*, pp. 214–223, 2017.

David Bau, Jun-Yan Zhu, Jonas Wulff, William S. Peebles, Bolei Zhou, Hendrik Strobelt, and Antonio Torralba. Seeing what a GAN cannot generate. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4501–4510, 2019.

Shiming Chen, Wenjie Wang, Beihao Xia, Qinmu Peng, Xinge You, Feng Zheng, and Ling Shao. FREE: feature refinement for generalized zero-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 122–131, 2021.

Shiming Chen, Ziming Hong, Wenjin Hou, Guo-Sen Xie, Yibing Song, Jian Zhao, Xinge You, Shuicheng Yan, and Ling Shao. Transzero++: Cross attribute-guided transformer for zero-shot learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):12844–12861, 2023a.

Shiming Chen, Wenjin Hou, Ziming Hong, Xiaohan Ding, Yibing Song, Xinge You, Tongliang Liu, and Kun Zhang. Evolving semantic prototype improves generative zero-shot learning. In *Proceedings of the International Conference on Machine Learning*, pp. 4611–4622, 2023b.

Shiming Chen, Wenjin Hou, Salman H. Khan, and Fahad Shahbaz Khan. Progressive semantic-guided vision transformer for zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23964–23974, 2024.

Shiming Chen, Dingjie Fu, Salman Khan, and Fahad Shahbaz Khan. Genzsl: Generative zero-shot learning via inductive variational autoencoder. In *Proceedings of the International Conference on Machine Learning*, 2025a.

Shiming Chen, Ziming Hong, Xinge You, and Ling Shao. Semantics-conditioned generative zero-shot learning via feature refinement. *International Journal of Computer Vision*, 133(7):4504–4521, 2025b.

Zhi Chen, Zecheng Zhao, Jingcai Guo, Jingjing Li, and Zi Huang. SVIP: semantically contextualized visual patches for zero-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3346–3356, 2025c.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pp. 2672–2680, 2014.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

Wenjin Hou, Shiming Chen, Shuhuang Chen, Ziming Hong, Yan Wang, Xuetao Feng, Salman H. Khan, Fahad Shahbaz Khan, and Xinge You. Visual-augmented dynamic semantic prototype for generative zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23627–23637, 2024.

Wenjin Hou, Xiaoxiao Sun, and Hehe Fan. Incentivizing generative zero-shot learning via outcome-reward reinforcement learning with visual cues. *arXiv preprint arXiv:2603.21138*, 2026.

Chenyi Jiang, Yuming Shen, Dubing Chen, Haofeng Zhang, Ling Shao, and Philip H. S. Torr. Estimation of near-instance-level attribute bottleneck for zero-shot learning. *International Journal of Computer Vision*, 132(8):2962–2988, 2024.

Huajie Jiang, Zhengxian Li, Xiaohan Yu, Yongli Hu, Baocai Yin, Jian Yang, and Yuankai Qi. Visual and semantic prompt collaboration for generalized zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20275–20285, 2025.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

Xia Kong, Zuodong Gao, Xiaofan Li, Ming Hong, Jun Liu, Chengjie Wang, Yuan Xie, and Yanyun Qu. En-compactness: Self-distillation embedding & contrastive generation for generalized zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9296–9305, 2022.

Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern*, pp. 951–958, 2009.

Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. In *Proceedings of the International Conference on Machine Learning*, pp. 1558–1566, 2016.

Xiaofan Li, Yachao Zhang, Shiran Bian, Yanyun Qu, Yuan Xie, Zhongchao Shi, and Jianping Fan. Vs-boost: Boosting visual-semantic association for generalized zero-shot learning. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 1107–1115, 2023.

Man Liu, Huihui Bai, Feng Li, Chunjie Zhang, Yunchao Wei, Tat-Seng Chua, and Yao Zhao. Attend and enrich: Enhanced visual prompt for zero-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 5504–5512, 2025a.

Man Liu, Huihui Bai, Feng Li, Chunjie Zhang, Yunchao Wei, Meng Wang, Tat-Seng Chua, and Yao Zhao. PSVMA+: exploring multi-granularity semantic-visual adaption for generalized zero-shot learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(1):51–66, 2025b.

Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. Unrolled generative adversarial networks. In *International Conference on Learning Representations*, 2017.

Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

Sanath Narayan, Akshita Gupta, Fahad Shahbaz Khan, Cees G. M. Snoek, and Ling Shao. Latent embedding feedback and discriminative features for zero-shot classification. In *Proceedings of the European Conference on Computer Vision*, pp. 479–495, 2020.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael G. Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Genevieve Patterson and James Hays. SUN attribute database: Discovering, annotating, and recognizing scene attributes. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2751–2758, 2012.

Scott E. Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descriptions. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 49–58, 2016.

Charles Spearman. The proof and measurement of association between two things. 1961.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

Zhengbo Wang, Jian Liang, Ran He, Nan Xu, Zilei Wang, and Tieniu Tan. Improving zero-shot generalization for CLIP with synthesized prompts. *arXiv preprint arXiv:2307.07397*, 2023.

Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 69–77, 2016.

Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5542–5551, 2018.

Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning - A comprehensive evaluation of the good, the bad and the ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(9):2251–2265, 2019a.

Yongqin Xian, Saurabh Sharma, Bernt Schiele, and Zeynep Akata. f-vaegan-d2: A feature generating framework for any-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10275–10284, 2019b.

Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion gans. In *International Conference on Learning Representations*, 2022.

Zihan Ye, Shreyank N. Gowda, Shiming Chen, Xiaowei Huang, Haotian Xu, Fahad Shahbaz Khan, Yaochu Jin, Kaizhu Huang, and Xiaobo Jin. Zerodiff: Solidified visual-semantic correlation in zero-shot learning. In *International Conference on Learning Representations*, 2025.

Li Zhang, Tao Xiang, and Shaogang Gong. Learning a deep embedding model for zero-shot learning. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 3010–3019. IEEE, 2017.

Zao Zhang, Liguo Sun, and Pin Lyu. SAGE: structured attribute-guided enhancement for GZSL. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 28564–28572, 2026.

A Appendix

A.1 Prototype Refinement Analysis

Prototype refinement in Eq. (5) aims to alleviate similarity compression while preserving the real inter-class structure. We investigate the effect of the suppression strength α on refined prototypes and downstream ZSL performance.

A larger α removes more shared representation, improving class separability but potentially distorting the real structure. As shown in Fig. 6(a), increasing α gradually decreases Spearman ρ and Retention@5, indicating increasing structural distortion. However, both metrics remain stable under moderate refinement, demonstrating that the refined prototypes learn the real class relationships.

Fig. 6(b) shows that increasing α reduces the mean similarity while increasing its standard deviation, indicating that refinement expands the dynamic range of visual similarity and alleviates similarity compression.

Fig. 6(c) further shows the impact of α on ZSL performance. Prototype refinement improves performance, validating its effectiveness. However, excessive refinement improves seen accuracy while degrading unseen accuracy, as over-suppression may distort visual class structures and lead VSA to learn biased structures that fit seen classes but fail to generalize to unseen classes. Therefore, we set $\alpha = 0.5$ to balance structural preservation and compression alleviation.

A.2 Hyper-parameter Analysis on AWA2 and SUN

We further provide sensitivity analyses of λ, λ_{SC}, and λ_{PA} on AWA2 and SUN in Fig. 7. Among them, λ controls the magnitude of AAD-induced semantic variations and directly affects the trade-off between semantic diversity and consistency. A small λ limits the utilization of learned attribute variations, resulting in insufficiently diverse semantic conditions and degraded unseen-class generation. In contrast, an excessively large λ introduces variations beyond the semantic prototype, causing performance degradation on both seen and unseen classes. It may even lead to overfitting to seen classes, as observed on AWA2 when $\lambda = 1.0$. These results further highlight the importance of properly controlling variation magnitude for effective semantic condition learning. Meanwhile, λ_{SC} and λ_{PA} remain stable over a wide range of values, demonstrating the robustness of our method.

A.3 Qualitative Analysis of Generated Feature

To qualitatively analyze the impact of each AAVS component on generated feature quality, we visualize and compare the feature distributions of f-VAEGAN (baseline), AAVS, f-VAEGAN+AAD, and f-VAEGAN+VSA using t-SNE (Van der Maaten & Hinton, 2008).

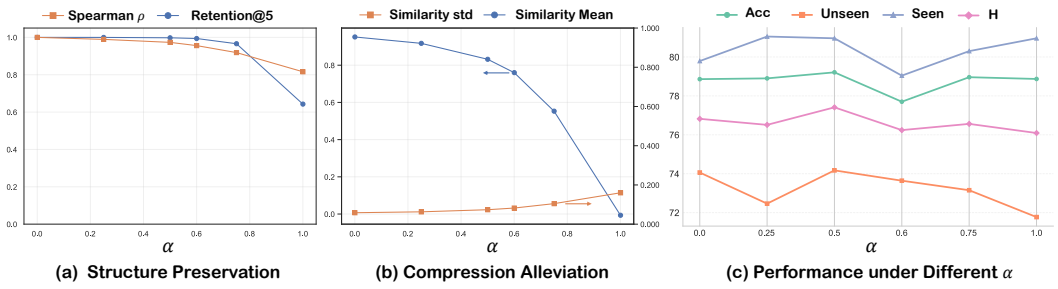

Figure 6: Analysis of the prototype refinement strength α on CUB. (a) Refinement preserves the original prototype structure measured by Spearman ρ and Retention@5. (b) Refinement alleviates similarity compression by reducing the mean similarity and increasing the similarity variance. (c) Moderate refinement improves ZSL performance, while excessive refinement causes structural deviation and degrades unseen-class generalization.

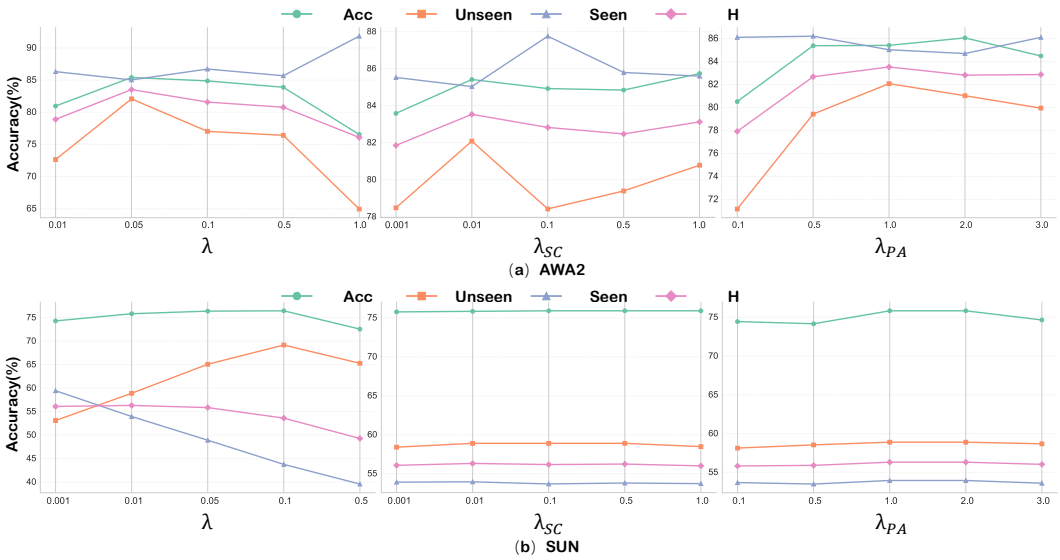

Figure 7: Hyper-parameter analysis on AWA2 and SUN. λ denotes the magnitude of learned variation in Eq. (2), λ_{SC} and λ_{PA} control the weights of two losses in Eq. (7), respectively.

Table 3: Dataset statistics under the Proposed Split (PS). s/u denotes the numbers of seen/unseen classes, and $|A|$ is the attribute dimension.

Dataset	**#Images**	**#Classes** (s/u)	$\lvert A\rvert$
AWA2 (Xian et al., 2019a)	37,322	50 (40/10)	85
CUB (Wah et al., 2011)	11,788	200 (150/50)	312
SUN (Patterson & Hays, 2012)	14,340	717 (645/72)	102

The baseline suffers from distribution shifts between generated and real features (red circles in Fig. 8(a)), as it inherits semantic structures inconsistent with the visual space. VSA mitigates this issue by introducing visually aligned attribute embeddings, producing features closer to the real distribution (green circles in Fig. 8(d)).

Moreover, static class-level attributes lead to concentrated generated features with limited intra-class diversity (**black circles** in Fig. 8(a)), while AAD improves diversity by learning adaptive attribute distributions (**black circles** in Fig. (c)).

Combining both components, AAVS generates features with improved diversity and better visual distribution alignment, as shown in Fig. 8(b).

A.4 Benchmark Datasets

We evaluate AAVS on three widely used ZSL benchmarks under the Proposed Split (PS) protocol (Xian et al., 2019a), as summarized in Table 3. AWA2 is a coarse-grained animal recognition dataset containing 37,322 images from 50 classes, each represented by an 85-dimensional attribute vector. CUB is a fine-grained bird dataset with 11,788 images from 200 species and 312-dimensional attributes. SUN is a large-scale scene recognition dataset comprising 14,340 images from 717 scene categories, described by 102-dimensional attributes.

Following PS, AWA2 is divided into 40 seen and 10 unseen classes, CUB into 150 seen and 50 unseen classes, and SUN into 645 seen and 72 unseen classes. Only images from seen classes are available for model training, while unseen-class images are reserved for evaluation. Under CZSL, classification is restricted to unseen classes; under GZSL, both seen and unseen classes constitute the test label space.

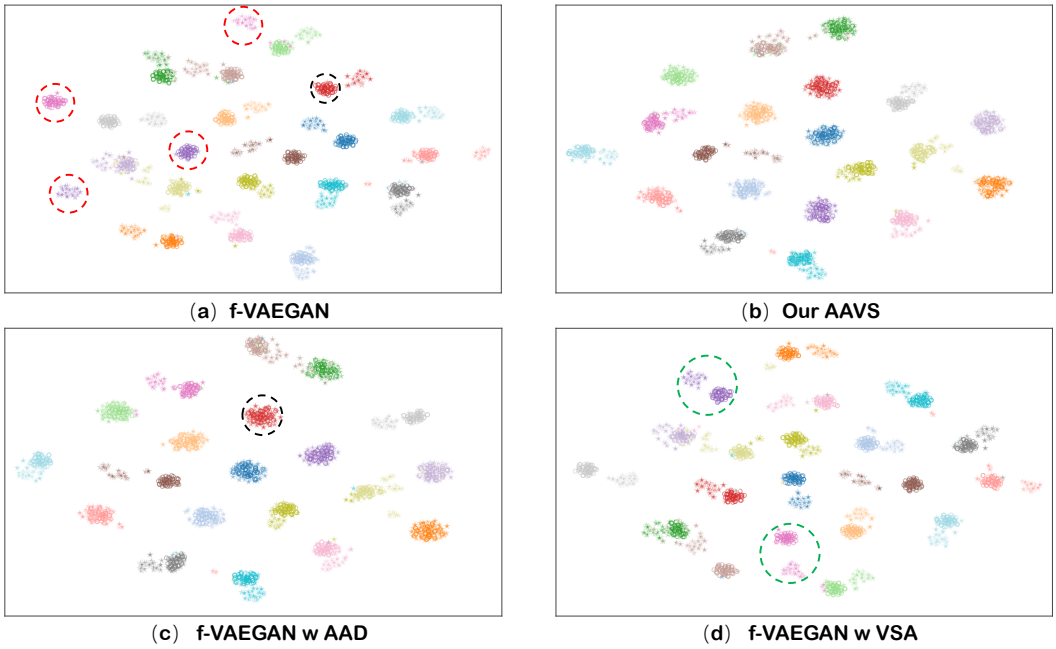

Figure 8: Qualitative evaluation of AAVS on CUB. We randomly select 10 seen and 10 unseen classes, with 50 generated and 50 real features sampled per class. Different colors indicate classes, while ○ and ⋆ denote generated and real features, respectively. (Zoom in for details.)

A.5 Full algorithm

Algorithms 1 and 2 summarize the training and evaluation procedures of AAVS.

During training, we construct refined prototypes and compute the inter-class discriminability. For each seen-class feature x_i^s, E^+ predicts an adaptive attribute distribution and samples $\tilde{a}_i$, while E^* maps it to v_i. Their concatenation forms the generation condition, and E^+, E^*, and the generator G are jointly optimized.

At inference, AAVS constructs conditions solely from unseen-class attributes and synthesizes N_{syn} features per unseen class using distinct semantic conditions. We then train the corresponding classifier and evaluate it on the test data under each ZSL setting.

Algorithm 1 Training AAVS and the Conditional Generator

Input: Seen-class training data $\mathcal{D}_{\text{tr}}^{s}$, class attributes $\{a_c\}_{c\in\mathcal{Y}^s}$, visual encoder F, number of training epochs T, and hyperparameters λ, λ_{SC}, and λ_{PA}
Output: Trained adaptive attribute encoder E^{+}, structure alignment encoder E^{*}, and generator G
Initialize E^{+}, E^{*}, and G
Extract seen-class visual features $x_i = F(I_i)$ and compute class prototypes $p_c = \frac{1}{N_c}\sum_{i:y_i=c} x_i$
Estimate the dominant common direction p_{com} and refine the prototypes as $\hat{p}_c = p_c - \alpha p_{\text{com}}$
Compute and normalize attribute discriminability $\{\bar{d}_j\}_{j=1}^{|A|}$ from the seen-class attribute matrix
for $t = 1$ to T **do**
  Sample a mini-batch $\mathcal{B} \subset \mathcal{D}_{\text{tr}}^{s}$
  for all $(x^s, a^s, y^s) \in \mathcal{B}$ **do**
    Predict attribute-specific parameters $\{(\mu_c^j, \sigma_c^j)\}_{j=1}^{|A|} \leftarrow \mathrm{E}^{+}(a_c)$
    Compute the calibrated scale $\tilde{\sigma}_c^j = (1-\bar{d}_j)\sigma_c^j$
    Sample attribute residuals $\Delta a_c^j \sim \mathcal{N}(\mu_c^j, (\tilde{\sigma}_c^j)^2)$
    Construct the adaptive attribute $\tilde{a}_c^j \leftarrow a_c^j + \lambda \Delta a_c^j$
    Obtain the visual-grounded embedding $v_c \leftarrow \mathrm{E}^{*}(\tilde{a}_c)$
    Generate visual features using the conditional VAE-GAN: $\hat{x} \leftarrow G(x^s, [\tilde{a}_c, v_c])$
  end for
  Compute the generator loss $\mathcal{L}_G$, semantic consistency loss $\mathcal{L}_{SC}$, and prototype alignment loss $\mathcal{L}_{PA}$
  Compute $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_G + \lambda_{SC}\mathcal{L}_{SC} + \lambda_{PA}\mathcal{L}_{PA}$
  Update E^{+}, E^{*}, and G by minimizing $\mathcal{L}_{\text{total}}$
end for
return E^{+}, E^{*}, and G

Algorithm 2 Unseen-Class Feature Synthesis and ZSL Evaluation

Input: Trained E^{+}, E^{*}, and G; unseen-class attributes $\{a_u\}_{u\in\mathcal{Y}^u}$; seen-class training features $\mathcal{X}_{\text{tr}}^{s}$; numbers of synthesized features N_{syn} and evaluation mode $\in \{\text{CZSL}, \text{GZSL}\}$
Output: Classification performance on unseen classes or on seen and unseen classes
Initialize the synthetic unseen-class set $\widetilde{\mathcal{D}}^{u} \leftarrow \emptyset$
for each unseen class $u \in \mathcal{Y}^u$ **do**
  Predict $\{(\mu_u^j, \sigma_u^j)\}_{j=1}^{|A|} \leftarrow \mathrm{E}^{+}(a_u)$
  Calibrate the variation scale using $\tilde{\sigma}_u^j = (1-\bar{d}_j)\sigma_u^j$
  for $n = 1$ to N_{syn} **do**
    Sample $\Delta a_u^j \sim \mathcal{N}(\mu_u^j, (\tilde{\sigma}_u^j)^2)$
    Construct $\tilde{a}_u^j \leftarrow a_u^j + \lambda \Delta a_u^j$
    Compute $v_u \leftarrow \mathrm{E}^{*}(\tilde{a}_u)$ and sample latent noise z
    Generate $\hat{x}_{u,n} \leftarrow G(z; [\tilde{a}_u, v_u])$
    Add $(\hat{x}_{u,n}, u)$ to $\widetilde{\mathcal{D}}^{u}$
  end for
end for
if mode $=$ CZSL **then**
  Set classifier training data $\mathcal{D}_{\text{cls}} \leftarrow \widetilde{\mathcal{D}}^{u}$
else
  Set classifier training data $\mathcal{D}_{\text{cls}} \leftarrow \mathcal{X}_{\text{tr}}^{s} \cup \widetilde{\mathcal{D}}^{u}$
end if
Train a classifier f on $\mathcal{D}_{\text{cls}}$
if mode $=$ CZSL **then**
  Evaluate f on unseen-class test data and report Acc
else
  Evaluate f on seen and unseen test data and report U, S, and H $= \frac{2\times \text{S}\times \text{U}}{\text{S}+\text{U}}$
end if