SEMANTIC PURIFICATION FOR CONDITIONAL REPRESENTATION LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Conditional representation learning aims to extract criterion-specific features for customized tasks. Recent methods construct conditional subspaces spanned by criterion-specific text bases in the embedding space of vision-language models (VLMs). Image embeddings are then projected onto these subspaces to obtain conditional representations. However, since VLMs are not explicitly trained to disentangle semantics associated with different criteria, the corresponding conditional subspaces remain coupled. This coupling induces semantic leakage during projection, thereby degrading the semantic purity of conditional representations. To suppress semantic leakage, we propose Semantic Purification for Conditional Representation Learning (SP-CRL). Specifically, SP-CRL first decomposes the original text basis and performs curvature-based adaptive truncation on the resulting basis vectors to construct a purer conditional subspace. It then identifies an appropriate noise subspace and projects image embeddings onto its null space to remove irrelevant semantic components. Extensive experiments across customized clustering, customized few-shot classification, and customized retrieval tasks demonstrate that SP-CRL achieves state-of-the-art performance with superior generalization. **The code is available in the supplementary material.**

1 INTRODUCTION

Representation learning is a fundamental task in machine learning that extracts meaningful abstractions from raw data. Enabled by techniques such as contrastive learning (Chen et al., 2020; He et al., 2020; Radford et al., 2021) and masked prediction (Devlin et al., 2019; He et al., 2022; Wei et al., 2022), representation learning has been successfully applied to multiple downstream tasks (Mo et al., 2022; Gordo et al., 2016; Sarafianos et al., 2019; Yan et al., 2023). However, existing representation learning methods do not explicitly balance distinct semantic factors. As a result, the learned representations tend to be dominated by salient semantic factors (e.g., category), while less salient factors (e.g., count) are overshadowed. As shown in Figure 1(d), this limitation constrains the applicability of representation learning to customized tasks with varying criteria.

Consequently, conditional representation learning has emerged as a promising paradigm that extracts criterion-specific features tailored to particular requirements. Existing methods can be broadly categorized into supervised, prompt-driven, and projection-based methods. Supervised methods (Geng et al., 2025; Liu et al., 2024b) learn conditional representations by retraining models with criterion-specific supervision signals, incurring prohibitive annotation costs. Prompt-driven methods (Kwon et al., 2024; Wang et al., 2026) elicit conditional representations from MLLMs via criterion-specific prompts, but repeated querying of MLLMs leads to substantial computational overhead.

Recently, projection-based methods (Yao et al., 2024a;b; Liu et al., 2026) have been employed to achieve efficient conditional representation learning. Specifically, LLM-generated descriptions are first encoded through a VLM's text encoder to construct criterion-specific text bases. Image embeddings are then projected onto the conditional subspaces spanned by these bases to obtain conditional representations. Unlike prompt-driven methods, projection-based methods invoke the LLM only once to construct criterion-specific text bases, avoiding repeated LLM queries at inference time.

However, existing projection-based methods focus on the conditional subspace corresponding to the target criterion, overlooking its coupling with other conditional subspaces. Since VLMs lack explicit semantic disentanglement constraints during training, text bases associated with different criteria

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.

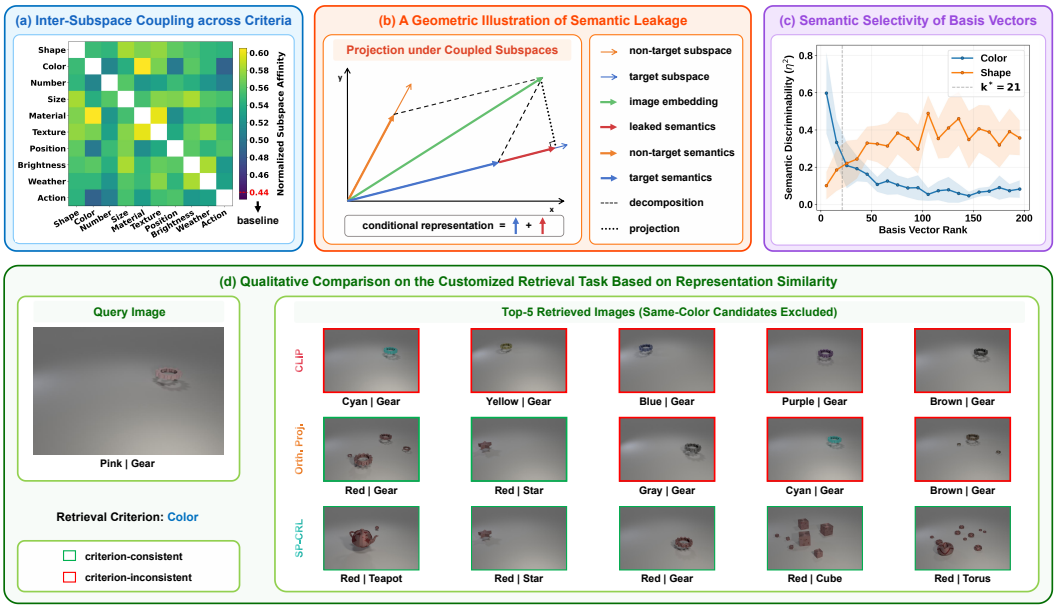


Figure 1: **(a) Inter-subspace coupling across criteria.** We use normalized subspace affinity (Soltanolkotabi et al., 2014) to quantify coupling between conditional subspaces. The red dashed line marks the expected affinity of randomly oriented subspaces with matched dimensionality. All pairwise affinities exceed this baseline, demonstrating that coupling is inherent and pervasive among conditional subspaces. **(b) A geometric illustration of semantic leakage.** Under coupled subspaces, the non-target semantic component possesses a non-zero projection onto the target subspace, introducing undesired semantics into the final conditional representation. **(c) Semantic selectivity of basis vectors.** As singular values decrease, basis vectors exhibit decreasing semantic discriminability for the target criterion (i.e., Color) but increasing semantic discriminability for the non-target criterion (i.e., Shape). The dashed line denotes the curvature-based truncation boundary. Further experimental details are provided in Appendix A. **(d) Qualitative comparison on the customized retrieval task based on representation similarity.** For color-based retrieval, CLIP (Radford et al., 2021) is dominated by shape semantics and retrieves only gear-shaped samples with colors substantially different from the query. Orthogonal projection reduces the influence of shape semantics but still retrieves several gear-shaped, color-dissimilar samples. In contrast, SP-CRL effectively suppresses semantic leakage, retrieving shape-diverse samples with colors more similar to the query.

may share semantic directions in the VLM embedding space. As illustrated in Figure 1(a), the resulting conditional subspaces consistently exhibit coupling, regardless of their semantic relatedness. Given such coupling, even standard orthogonal projection is insufficient to obtain semantically pure conditional representations: non-target semantic components possess non-zero projections onto the target subspace (Figure 1(b)). We refer to this phenomenon as semantic leakage, as it introduces undesired semantics into the final conditional representations, compromising their semantic purity.

In this paper, we propose Semantic Purification for Conditional Representation Learning (SP-CRL), which consists of two components: Criterion-Specific Basis Refinement (CSBR) and Null-Space Denoising Projection (NSDP). CSBR employs singular value decomposition to decompose the LLM-generated text basis into a set of orthonormal basis vectors, ranked by their singular values. As illustrated in Figure 1(c), these basis vectors exhibit varying semantic selectivity: those associated with larger singular values show higher semantic discriminability for the target criterion, while those with smaller singular values are relatively more discriminative for the non-target criterion. Motivated by this observation, CSBR applies curvature-based adaptive truncation to retain criterion-specific basis vectors, which span a purer conditional subspace with reduced semantic leakage. To further suppress residual semantic leakage, NSDP identifies an appropriate noise subspace by greedily incorporating non-target subspaces based on the potential denoising gain and loss. It then projects the image embeddings onto the null space of the noise subspace, explicitly removing irrelevant semantic components prior to conditional representation extraction. Both the qualitative

retrieval visualization (Figure 1(d)) and the quantitative linear probing analysis (Appendix B) show that SP-CRL effectively suppresses semantic leakage while largely preserving target semantics.

In summary, our contributions are as follows:

- We reveal that conditional subspaces associated with different criteria exhibit inherent and pervasive coupling. Such coupling induces semantic leakage during projection, thereby degrading the semantic purity of conditional representations.
- We propose Semantic Purification for Conditional Representation Learning, a projection-based conditional representation learning framework that suppresses semantic leakage through criterion-specific basis refinement and null-space denoising projection.
- Extensive experiments show that SP-CRL achieves state-of-the-art performance on diverse tasks, including customized clustering, customized few-shot classification, and customized retrieval.

2 Related Work

2.1 General Representation Learning

General representation learning aims to extract general-purpose features from raw data. As a classical unsupervised learning method, autoencoders (Hinton & Salakhutdinov, 2006) obtain compact representations by minimizing reconstruction error. Denoising autoencoders (Vincent et al., 2008) and variational autoencoders (Kingma & Welling, 2013) introduce corruption robustness and probabilistic modeling respectively, improving latent space stability and structure. In recent years, the self-supervised learning paradigm has expanded the boundaries of representation learning. Pretext tasks such as patch and rotation prediction (Doersch et al., 2015; Gidaris et al., 2018), solving jigsaw puzzles (Noroozi & Favaro, 2016), and colorization (Zhang et al., 2016) enable models to capture global-local consistency. More recently, contrastive learning frameworks (Chen et al., 2020; He et al., 2020) have employed instance discrimination to learn highly discriminative representations without manual annotation. Vision-language models (Radford et al., 2021; Li et al., 2023a; Zhai et al., 2023) also employ image-text contrastive objectives to align visual and textual representations. However, existing methods tend to overemphasize salient semantic factors while underrepresenting less salient ones, limiting their applicability to customized tasks with varying criteria.

2.2 Conditional Representation Learning

Conditional representation learning aims to extract criterion-specific features from raw data. Supervised methods (Geng et al., 2025; Liu et al., 2024b) provide a straightforward solution by retraining models with criterion-specific supervision signals. However, this approach is fundamentally constrained by prohibitive annotation costs. To mitigate reliance on supervision signals, recent methods leverage MLLMs to elicit conditional representations through carefully designed prompts. IC|TC (Kwon et al., 2024) obtains conditional labels for images via direct dialogue with MLLMs, while ESMC (Wang et al., 2026) selects intermediat hidden states of MLLMs that are most responsive to the target criterion as conditional representations. Nevertheless, the substantial computational overhead caused by repeated MLLM invocations during inference remains a significant bottleneck. Projection-based methods obtain conditional representations by projecting image embeddings onto subspaces spanned by criterion-specific text bases. Multi-Map (Yao et al., 2024b) and Multi-Sub (Yao et al., 2024a) learn a set of trainable text basis weights for each image, which encode criterion-relevant semantics within the conditional subspace. CRL (Liu et al., 2026) directly employs cosine similarity between image embeddings and text basis vectors as conditional representations. However, geometric coupling among conditional subspaces induces semantic leakage during projection, degrading the semantic purity of conditional representations.

3 Method

3.1 Formulation and Framework

Given an image set $\mathcal{X}$ and its corresponding criterion set $\mathcal{C}$, conditional representation learning aims to extract representations aligned with a user-specified target criterion $C_t \in \mathcal{C}$.

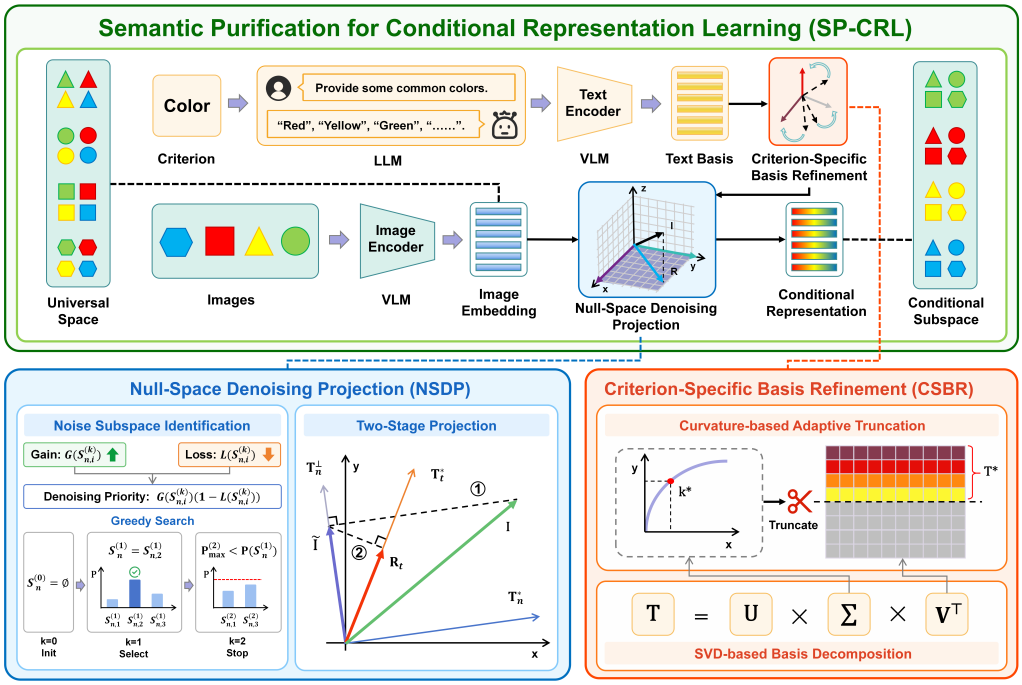


Figure 2: The framework of SP-CRL. In CSBR, the original text basis is decomposed via SVD into $\mathbf{U}$, $\mathbf{\Sigma}$, and $\mathbf{V}^\top$, where the top k^* right singular vectors are retained as the refined orthogonal basis $\mathbf{T}^*$. The truncation boundary k^* is determined by the maximum curvature of the cumulative energy curve. In NSDP, the candidate noise subspace with the highest denoising priority is identified as the appropriate noise subspace $\mathcal{S}_n$. The image embeddings $\mathbf{I}$ are first projected onto the null space of the identified noise subspace to obtain the denoised embeddings $\tilde{\mathbf{I}}$ (①), which are then projected onto the target subspace to extract the conditional representations $\mathbf{R}_t$ (②).

As shown in Figure 2, SP-CRL projects image embeddings onto the conditional subspace to obtain conditional representations. To construct the basis of this subspace, we query an LLM to generate criterion-relevant descriptive text W:

$$W = \text{LLM}(P_1, C_t), \tag{1}$$

where P_1 denotes the LLM prompt template conditioned on criterion C_t.

The generated W is then encoded by a VLM to obtain a set of basis vectors $\mathbf{T}$, which span the conditional subspace corresponding to criterion C_t:

$$\mathbf{T} = \text{VLM}_{\text{text}}(P_2, W), \tag{2}$$

where P_2 denotes the VLM prompt template that is concatenated with W to construct a complete sentence. More details about the prompt templates P_1 and P_2 can be found in Appendix C.

In parallel, the images in $\mathcal{X}$ are encoded through the same VLM to extract their embeddings $\mathbf{I}$:

$$\mathbf{I} = \text{VLM}_{\text{image}}(\mathcal{X}). \tag{3}$$

Prior to projection, both the image embeddings $\mathbf{I}$ and the basis vectors $\mathbf{T}$ are ℓ_2-normalized. The conditional representations $\mathbf{R}$ are then obtained by projecting the image embeddings $\mathbf{I}$ onto the conditional subspace spanned by $\mathbf{T}$:

$$\mathbf{R} = \mathbf{I}\mathbf{T}^\top \left(\mathbf{T}\mathbf{T}^\top\right)^\dagger \mathbf{T}, \tag{4}$$

where symbol † denote the Moore–Penrose pseudoinverse.

To suppress semantic leakage, SP-CRL introduces two complementary components: Criterion-Specific Basis Refinement (CSBR) and Null-Space Denoising Projection (NSDP), which are detailed in the following sections.

3.2 CRITERION-SPECIFIC BASIS REFINEMENT

Existing methods encode LLM-generated descriptions into text embeddings and directly employ them as bases for conditional subspaces. However, such text bases capture not only criterion-relevant semantics but also irrelevant semantics. To mitigate this issue, we introduce Criterion-Specific Basis Refinement, which first decomposes the original text basis into orthogonal basis vectors and then adaptively discards those dominated by irrelevant semantics.

SVD-based Basis Decomposition. Given the original text basis $\mathbf{T} \in \mathbb{R}^{n\times d}$, where n denotes the number of descriptive texts and d represents the embedding dimension, we perform SVD:

$$\mathbf{T} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top, \tag{5}$$

where $\mathbf{U} \in \mathbb{R}^{n\times n}$ and $\mathbf{V} \in \mathbb{R}^{d\times d}$ are column-orthogonal matrices, and $\mathbf{\Sigma} \in \mathbb{R}^{n\times d}$ is a diagonal matrix containing the singular values in descending order.

The column vectors of $\mathbf{V}$ form an orthonormal basis for the conditional subspace. The orthogonality enables each basis vector to be independently analyzed and filtered based on its semantic selectivity.

Curvature-based Adaptive Truncation. Our empirical analysis (Appendix A) reveals that basis vectors associated with larger singular values are more criterion-specific, with higher discriminability for criterion-relevant semantics and lower discriminability for irrelevant semantics. This semantic selectivity progressively diminishes as the singular value decreases. Based on this observation, we identify the turning point in the singular-value energy distribution and truncate the basis vectors accordingly. Specifically, we first compute the cumulative energy ratio:

$$E(k) = \frac{\sum_{i=1}^{k} \sigma_i^2}{\sum_{i=1}^{n} \sigma_i^2}, \tag{6}$$

where σ_i denotes the i-th singular value. The cumulative energy curve $E(k)$ typically exhibits an "elbow" shape: it rises steeply as principal components capture most variance, then flattens as subsequent components contribute diminishing information.

To quantify this "elbow" behavior, we construct a normalized discrete curve (x_i, y_i) for $i = 1, 2, \ldots, n$, where $x_i = (i-1)/(n-1) \in [0, 1]$ represents the normalized proportion of retained basis vectors, and y_i is the min-max normalized cumulative energy derived from $E(k)$:

$$y_i = \frac{E(i) - E(1)}{E(n) - E(1)} \in [0, 1]. \tag{7}$$

The curvature at each point is then computed as:

$$\kappa(x_i) = \frac{|y''(x_i)|}{(1 + (y'(x_i))^2)^{3/2}}, \tag{8}$$

where $y'(x_i)$ and $y''(x_i)$ are estimated via finite differences.

The adaptive truncation boundary k^* is determined by the maximum-curvature point:

$$k^* = \arg\max_i \kappa(x_i). \tag{9}$$

The maximum-curvature point marks where the marginal energy gain begins to diminish, signaling the transition from criterion-specific basis vectors to those dominated by irrelevant semantics.

Finally, we construct the refined basis $\mathbf{T}^*$ by retaining the top k^* right singular vectors:

$$\mathbf{T}^* = (\mathbf{V}_{:,1:k^*})^\top \in \mathbb{R}^{k^*\times d}. \tag{10}$$

3.3 NULL-SPACE DENOISING PROJECTION

CSBR refines the text basis to construct a purer conditional subspace, suppressing semantic leakage at the subspace level. However, due to the inherent coupling between subspaces, residual semantic leakage may still persist. We therefore propose Null-Space Denoising Projection to further suppress semantic leakage at the projection level. NSDP first identifies an appropriate noise subspace, and then projects image embeddings onto its null space to remove irrelevant semantic components.

Noise Subspace Identification. Although all non-target subspaces may introduce semantic leakage, indiscriminately merging multiple of them into the noise subspace is not necessarily beneficial. A broader noise subspace increases the risk of removing target semantic components during denoising projection. Therefore, we jointly consider the potential denoising gain and loss, and employ a greedy search strategy to identify an appropriate noise subspace.

Let $\mathcal{N}^{(0)}$ denote the index set of all non-target criteria in $\mathcal{C}$. We initialize the noise subspace as $\mathcal{S}_n^{(0)} = \emptyset$. At the k-th iteration, each remaining non-target subspace is combined with the noise subspace selected in the previous iteration to form a candidate:

$$\mathcal{S}_{n,i}^{(k)} = \mathcal{S}_n^{(k-1)} + \mathcal{S}_i, \quad i \in \mathcal{N}^{(k-1)}, \tag{11}$$

where $\mathcal{N}^{(k-1)}$ denotes the index set of the remaining non-target criteria. For each candidate, we concatenate the bases of the involved subspaces and apply CSBR to obtain the refined basis $\mathbf{T}_{n,i}^*$.

First, we quantify the potential denoising gain as the proportion of image-embedding energy captured by each candidate noise subspace:

$$G(\mathcal{S}_{n,i}^{(k)}) = \frac{\left\|\mathbf{I}(\mathbf{T}_{n,i}^*)^\top \mathbf{T}_{n,i}^*\right\|_F^2}{\|\mathbf{I}\|_F^2}, \tag{12}$$

where $\|\cdot\|_F$ denotes the Frobenius norm.

Second, we quantify the potential denoising loss as the proportion of target-subspace energy captured by each candidate noise subspace:

$$L(\mathcal{S}_{n,i}^{(k)}) = \frac{\left\|\mathbf{T}_t^*(\mathbf{T}_{n,i}^*)^\top \mathbf{T}_{n,i}^*\right\|_F^2}{\|\mathbf{T}_t^*\|_F^2}, \tag{13}$$

where $\mathbf{T}_t^*$ denotes the refined basis of the target subspace.

To balance semantic leakage suppression and target semantic preservation, we define the denoising priority of each candidate noise subspace as:

$$P(\mathcal{S}_{n,i}^{(k)}) = G(\mathcal{S}_{n,i}^{(k)}) \left(1 - L(\mathcal{S}_{n,i}^{(k)})\right). \tag{14}$$

At each iteration, the candidate with the highest denoising priority is greedily selected as the current noise subspace:

$$\mathcal{S}_n^{(k)} = \arg\max_{\mathcal{S}_{n,i}^{(k)}} P(\mathcal{S}_{n,i}^{(k)}). \tag{15}$$

The search continues while the denoising priority improves; otherwise, it terminates and takes the noise subspace selected in the previous iteration as the final noise subspace $\mathcal{S}_n$.

Note that the above procedure presumes non-target criteria are specified in advance. For practical scenarios where non-target criteria are unavailable, we provide an MLLM-based procedure in Appendix D to automatically infer them for noise subspace identification.

Null-Space Projection. After identifying the noise subspace, we project the image embeddings $\mathbf{I}$ onto its null space to remove the corresponding semantic components:

$$\tilde{\mathbf{I}} = \mathbf{I} - \mathbf{I}(\mathbf{T}_n^*)^\top \mathbf{T}_n^*. \tag{16}$$

where $\mathbf{T}_n^*$ denotes the refined basis of the identified noise subspace.

Conditional Representation Extraction. Finally, we extract the conditional representations by projecting the denoised image embeddings $\tilde{\mathbf{I}}$ onto the target subspace:

$$\mathbf{R}_t = \tilde{\mathbf{I}}(\mathbf{T}_t^*)^\top \mathbf{T}_t^*. \tag{17}$$

4 EXPERIMENTS

To evaluate the performance of the proposed SP-CRL, we apply the method to three customized tasks, including clustering, few-shot classification, and retrieval. Across these tasks, we employ diverse evaluation criteria and feature-rich datasets to enable a comprehensive assessment. We further conduct ablation studies, visualize representations (Appendix E), and perform sensitivity analyses on basis quantity (Appendix F.1), LLM selection (Appendix F.2) and VLM selection (Appendix F.3) to demonstrate the effectiveness and robustness of the proposed method.

4.1 CUSTOMIZED CLUSTERING

Dataset. For this task, we employ Clevr4-10k (Vaze et al., 2023), Cards (Yao et al., 2023) and Flowers as benchmark datasets. Clevr4-10k is a synthetic dataset containing 10,531 samples with four distinct partition criteria: "Shape", "Texture", "Color", and "Count". Cards is a poker card dataset comprising 8,029 samples partitioned according to two criteria: "Number" and "Suit". Moreover, we construct a new real-world flower dataset, Flowers, based on the Flowers102 dataset (Nilsback & Zisserman, 2008), which consists of 533 samples annotated according to two criteria: "Species" and "Color". The detailed information of the datasets can be found in Appendix C.

Setup. We use a pre-trained CLIP (Radford et al., 2021) model to encode images and LLM-generated text, from which SP-CRL derives conditional representations. We then perform k-means clustering on these representations, repeating the procedure 20 times with different random seeds, and report the average results. We employ three widely used clustering metrics for evaluation: Normalized Mutual Information (NMI), Accuracy (ACC), and Adjusted Rand Index (ARI).

Performance. As shown in Table 1, SP-CRL significantly surpasses all other customized clustering methods across all criteria, including TAC (Li et al., 2023b), PCL (Geng et al., 2025), Multi-Map (Yao et al., 2024b) and Multi-Sub (Yao et al., 2024a). When using CLIP as the backbone, SP-CRL achieves an average performance improvement of 22.21% over CRL (Liu et al., 2026) across all evaluation metrics. Moreover, it achieves strong performance even for the shape criterion, where traditional clustering methods such as SCAN (Van Gansbeke et al., 2020) typically perform best. Remarkably, SP-CRL accomplishes this without requiring any training, effectively balancing efficiency with superior performance.

Table 1: Performance on the task of customized clustering.

Method	Clevr4-10k												Mean
	Texture			Shape			Color			Count			
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	
SCAN	0.41	11.97	0.86	**90.99**	89.10	84.03	0.20	11.51	0.01	3.42	14.29	1.23	25.67
TAC	9.61	22.89	5.94	79.07	87.13	75.23	76.63	78.70	67.49	22.05	24.44	11.05	46.69
PCL	9.27	20.03	4.30	70.25	74.71	61.51	74.36	66.93	58.71	27.19	29.06	14.12	42.54
Multi-Map	3.77	17.25	1.81	67.48	66.01	57.40	56.83	56.46	45.73	11.38	20.13	7.67	34.33
Multi-Sub	4.93	18.30	3.52	72.48	72.07	61.48	64.34	62.19	55.62	13.77	21.64	9.28	38.30
CLIP	1.11	13.09	0.41	74.22	73.19	64.15	0.83	12.23	0.27	9.50	19.02	5.70	22.81
CRL	10.91	25.22	6.55	78.40	81.83	71.35	88.09	87.26	81.07	25.51	26.29	12.52	49.58
SP-CRL	**14.27**	**28.27**	**9.53**	86.67	**90.69**	**84.11**	**89.99**	**89.88**	**85.09**	**28.50**	**29.82**	**15.54**	**54.36**

Method	Cards						Flowers						Mean
	Number			Suit			Species			Color			
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	
SCAN	11.11	18.21	17.60	15.01	32.02	9.48	53.05	78.04	47.08	38.12	56.66	30.03	33.87
TAC	17.34	24.17	9.92	35.18	63.06	32.46	99.19	99.81	99.60	47.10	54.97	33.05	51.32
PCL	29.00	32.61	16.18	38.58	60.18	33.89	79.72	93.43	85.18	42.77	65.10	41.19	51.49
Multi-Map	16.32	20.61	7.95	14.02	46.65	11.08	97.70	97.80	96.89	50.96	60.34	38.58	46.57
Multi-Sub	19.58	23.27	10.28	18.76	49.09	15.24	98.35	98.84	98.52	53.17	64.21	42.92	49.35
CLIP	16.84	18.91	8.44	16.52	43.74	12.93	93.88	91.19	89.67	32.74	49.64	19.19	41.14
CRL	24.68	28.59	12.27	37.70	62.08	34.36	99.29	99.16	99.10	57.68	68.94	48.66	56.04
SP-CRL	**38.78**	**44.45**	**28.40**	**41.61**	**73.22**	**42.20**	**100.0**	**100.0**	**100.0**	**63.47**	**79.93**	**59.57**	**64.30**

4.2 CUSTOMIZED FEW-SHOT CLASSIFICATION

Dataset. Similar to customized clustering, we continue to use the Clevr4-10k, Cards and Flowers datasets to evaluate SP-CRL's performance on customized few-shot classification.

Setup. Consistent with Section 4.1, CLIP is used to encode both images and text. We employ logistic regression from scikit-learn for few-shot learning with 1, 5, and 10 samples per class. For SP-CRL, the denoising gain is quantified using only the training split to ensure a fair comparison. We repeat each setting over 20 random training splits and report the average accuracy.

Performance. Table 2 demonstrates the effectiveness of SP-CRL on customized few-shot classification. Compared to vanilla CLIP, SP-CRL achieves substantial improvements, with an average gain of 47.73% across all settings. Against the strong baseline CRL, SP-CRL consistently delivers superior performance, particularly when training examples are extremely limited. Notably, for color-based classification on Clevr4-10k, SP-CRL achieves a 25.44% improvement over CRL in the 1-shot setting. These consistent performance gains confirm SP-CRL's ability to extract discriminative conditional representations that effectively align with specific criteria.

Table 2: Performance on the task of customized few-shot classification.

Method	Clevr4-10k						Cards			Flowers			Mean
	Shape			Color			Number			Color			
	1	5	10	1	5	10	1	5	10	1	5	10	
CLIP	58.16	83.17	89.47	26.85	57.33	70.00	20.63	33.73	41.84	42.48	71.73	81.84	56.44
CRL	58.69	86.63	92.28	65.71	89.13	93.21	17.65	44.51	51.08	54.55	80.40	87.62	68.46
SP-CRL	**68.81**	**90.26**	**94.83**	**82.43**	**91.43**	**93.42**	**33.50**	**49.55**	**54.28**	**72.23**	**86.34**	**90.56**	**75.64**

4.3 Customized Retrieval

Dataset. Following prior work (Ma et al., 2020), we use DeepFashion (Liu et al., 2016) as the evaluation dataset for this task. The dataset comprises 221k, 27k, and 27k images for training, validation, and testing. This benchmark encompasses five criteria: "texture", "fabric", "shape", "part", and "style", containing 156, 218, 180, 216, and 230 distinct values.

Setup. For each query sample, we compute its cosine similarity with all candidate samples. The candidate samples are then ranked in descending order of similarity, and retrieval performance is evaluated accordingly. In the training-free setting, the conditional representations are directly evaluated. In the training setting, a two-layer MLP is additionally trained with triplet loss to further optimize the representations before evaluation. For fair comparison, SP-CRL quantifies the denoising gain using only the training split in both settings. We adopt Mean Average Precision (mAP) as the evaluation metric.

Table 3: Performance on the task of customized retrieval. Symbol ‡ signifies that training is conducted.

Method	Texture	Fabric	Shape	Part	Style	Mean
Triplet	13.26	6.28	9.49	4.43	3.33	7.36
CSN	14.09	6.39	11.07	5.13	3.49	8.01
ASEN	15.13	7.11	12.39	5.51	3.56	8.74
ASEN++	15.60	7.67	14.31	6.60	4.07	9.64
RPF	15.62	8.30	15.02	**7.38**	4.77	10.22
CLIP	9.14	4.68	7.86	4.26	4.48	6.08
CRL	11.03	6.76	11.80	5.56	4.42	7.93
CRL‡	15.92	8.54	16.33	6.90	5.27	10.59
SP-CRL	11.05	6.74	12.47	5.69	4.95	8.18
SP-CRL‡	**16.04**	**8.67**	**16.62**	7.14	**5.81**	**10.84**

Performance. Table 3 presents the performance comparison on customized retrieval. Equipped with merely a simple MLP, SP-CRL achieves the best mean mAP of 10.84%. This result outperforms previous methods, including Triplet (Veit et al., 2017), CSN (Veit et al., 2017), ASEN (Ma et al., 2020), ASEN++ (Dong et al., 2021), RPF (Dong et al., 2023) and CRL. Without training, SP-CRL already surpasses vanilla CLIP by 34.54% in mean mAP, validating its effectiveness in generating discriminative criterion-specific representations.

4.4 Ablation Study

To evaluate SP-CRL and its key design choices, we conduct ablation studies on the customized clustering task using the Clevr4-10k and Cards datasets. We examine component effectiveness, basis truncation in CSBR, and noise subspace identification in NSDP.

Ablation on Components. As shown in Table 4, we use standard orthogonal projection as the baseline to evaluate the contributions of CSBR and NSDP. On the Clevr4-10k dataset, CSBR alone improves the color-based clustering NMI from 73.91% to 87.87% by constructing a more criterion-specific basis, while NSDP alone increases the NMI to 78.95% by removing irrelevant semantic components in the identified noise subspace. Combining CSBR and NSDP further improves the NMI to 89.99%, achieving the best performance. Similar trends are observed on the Cards dataset, confirming the complementary effects of CSBR and NSDP.

Table 4: Ablation study on components.

Components		Clevr4-10k						Cards					
CSBR	NSDP	Shape			Color			Number			Suit		
		NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI
		81.72	82.12	75.00	73.91	70.78	63.99	36.86	39.38	23.27	35.99	55.41	28.44
✓		82.62	85.65	77.85	87.87	85.06	80.35	38.10	40.76	24.08	39.65	70.34	39.00
	✓	82.48	82.16	76.14	78.95	75.68	68.91	37.33	39.97	25.78	39.39	58.89	33.18
✓	✓	**86.67**	**90.69**	**84.11**	**89.99**	**89.88**	**85.09**	**38.78**	**44.45**	**28.40**	**41.61**	**73.22**	**42.20**

Ablation on Basis Truncation. We compare the proposed curvature-based adaptive truncation with fixed-ratio basis truncation. As shown in Table 5, the optimal truncation ratio (i.e., the proportion of retained basis vectors) varies across different criteria, while our strategy achieves consistently strong performance. This demonstrates the advantage of our strategy in identifying an appropriate truncation boundary for different criteria.

Table 5: Ablation study on basis truncation.

Retention Ratio	Clevr4-10k						Cards					
	Shape			Color			Number			Suit		
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI
20%	81.38	85.38	76.40	86.86	84.63	79.28	22.57	26.95	11.85	38.70	67.90	37.43
40%	81.99	84.70	77.05	85.10	82.08	76.55	31.74	34.72	18.30	35.82	58.23	31.16
60%	81.89	82.57	75.68	80.85	76.96	70.88	38.04	40.58	23.53	37.09	57.86	30.77
80%	**82.78**	84.08	77.19	76.63	72.55	66.07	37.85	40.66	23.86	36.05	57.51	30.08
100% (Full)	81.72	82.12	75.00	73.91	70.78	63.99	36.86	39.38	23.27	35.99	55.41	28.44
Adaptive (Ours)	82.62	**85.65**	**77.85**	**87.87**	**85.06**	**80.35**	**38.10**	**40.76**	**24.08**	**39.65**	**70.34**	**39.00**

Ablation on Noise Subspace Identification. We compare different noise subspace choices, including each non-target subspace individually and their combination, against a no-denoising baseline. Symbol * denotes the noise subspace identified by our method. As shown in Table 6, denoising with an inappropriate noise subspace can degrade the quality of conditional representations (e.g., using Shape as the noise subspace when Texture is the target subspace). In contrast, our identified noise subspace achieves the best performance, indicating a better balance between suppressing semantic leakage and preserving target semantics.

Table 6: Ablation study on noise subspace identification.

Noise Subspace	Clevr4-10k (Target Subspace)											
	Texture			Shape			Color			Count		
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI
None	13.80	27.86	8.60	82.62	85.65	77.85	87.87	85.06	80.35	25.74	28.56	14.22
Texture	-	-	-	82.55	84.16	76.92	87.97	88.28	82.04	26.38	29.33	14.88
Shape	10.45	24.75	6.05	-	-	-	**89.99***	**89.88***	**85.09***	**29.00**	29.16	15.01
Color	**14.27***	**28.27***	**9.53***	**86.67***	**90.69***	**84.11***	-	-	-	26.61	27.49	13.28
Count	11.23	26.00	7.19	81.51	83.22	75.94	87.94	86.67	81.42	-	-	-
Combination	10.95	23.59	6.35	84.89	87.63	80.66	89.33	87.15	83.09	28.50*	**29.82***	**15.54***

5 Conclusion

In this paper, we reveal that conditional subspaces corresponding to different criteria consistently exhibit coupling. Such coupling leads to semantic leakage during projection, degrading the semantic purity of conditional representations. To address this issue, we propose Semantic Purification for Conditional Representation Learning (SP-CRL), which consists of Criterion-Specific Basis Refinement (CSBR) and Null-Space Denoising Projection (NSDP). Extensive experiments across diverse customized tasks, including clustering, few-shot classification, and retrieval, demonstrate that SP-CRL consistently achieves superior performance with superior generalization.

AI USE STATEMENT

In this work, we used generative AI tools to assist with translation. We have not used generative AI tools for other tasks with required disclosure, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for brainstorming and editing the manuscript to improve readability and language quality. We have reviewed all AI-assisted work to ensure its accuracy, consistency with our intended meaning, and alignment with the actual research methodology and results. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision*, pp. 1422–1430, 2015.

Jianfeng Dong, Zhe Ma, Xiaofeng Mao, Xun Yang, Yuan He, Richang Hong, and Shouling Ji. Fine-grained fashion similarity prediction by attribute-specific embedding learning. *IEEE Transactions on Image Processing*, 30:8410–8425, 2021.

Jianfeng Dong, Xiaoman Peng, Zhe Ma, Daizong Liu, Xiaoye Qu, Xun Yang, Jixiang Zhu, and Baolong Liu. From region to patch: Attribute-aware foreground-background contrastive learning for fine-grained fashion retrieval. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1273–1282, 2023.

Xiwen Geng, Suyun Zhao, Yixin Yu, Borui Peng, Pan Du, Hong Chen, Cuiping Li, and Mengdie Wang. Personalized clustering via targeted representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 16790–16798, 2025.

Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

Albert Gordo, Jon Almazán, Jerome Revaud, and Diane Larlus. Deep image retrieval: Learning global representations for image search. In *European conference on computer vision*, pp. 241–257. Springer, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 9726–9735. IEEE, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 15979–15988. IEEE, 2022.

Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Sehyun Kwon, Jaden Park, Minkyu Kim, Jaewoong Cho, Ernest K Ryu, and Kangwook Lee. Image clustering conditioned on text criteria. In *International Conference on Learning Representations*, volume 2024, pp. 1393–1432, 2024.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023a.

Yunfan Li, Peng Hu, Dezhong Peng, Jiancheng Lv, Jianping Fan, and Xi Peng. Image clustering with external guidance. *arXiv preprint arXiv:2310.11989*, 2023b.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024a.

Honglin Liu, Peng Hu, Changqing Zhang, Yunfan Li, and Xi Peng. Interactive deep clustering via value mining. *Advances in Neural Information Processing Systems*, 37:42369–42387, 2024b.

Honglin Liu, Chao Sun, Peng Hu, Yunfan Li, and Xi Peng. Conditional representation learning for customized tasks. *Advances in Neural Information Processing Systems*, 38:31706–31737, 2026.

Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1096–1104, 2016.

Zhe Ma, Jianfeng Dong, Zhongzi Long, Yao Zhang, Yuan He, Hui Xue, and Shouling Ji. Fine-grained fashion similarity learning by attribute-specific embedding network. In *Proceedings of the AAAI Conference on artificial intelligence*, volume 34, pp. 11741–11748, 2020.

Yujie Mo, Liang Peng, Jie Xu, Xiaoshuang Shi, and Xiaofeng Zhu. Simple unsupervised graph representation learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 7797–7805, 2022.

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.

Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European conference on computer vision*, pp. 69–84. Springer, 2016.

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.

Nikolaos Sarafianos, Xiang Xu, and Ioannis A Kakadiaris. Adversarial representation learning for text-to-image matching. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5814–5824, 2019.

Mahdi Soltanolkotabi, Ehsan Elhamifar, and Emmanuel J Candes. Robust subspace clustering. 2014.

Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *European conference on computer vision*, pp. 268–285. Springer, 2020.

Sagar Vaze, Andrea Vedaldi, and Andrew Zisserman. No representation rules them all in category discovery. *Advances in Neural Information Processing Systems*, 36:19962–19989, 2023.

Andreas Veit, Serge Belongie, and Theofanis Karaletsos. Conditional similarity networks. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 1781–1789. IEEE, 2017.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pp. 1096–1103, 2008.

Xinyue Wang, Yuheng Jia, Hui Liu, and Junhui Hou. Esmc: mllm-based embedding selection for explainable multiple clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 26588–26596, 2026.

Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 14648–14658. IEEE, 2022.

Siming Yan, Zhenpei Yang, Haoxiang Li, Chen Song, Li Guan, Hao Kang, Gang Hua, and Qixing Huang. Implicit autoencoder for point-cloud self-supervised representation learning. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14484–14496. IEEE, 2023.

Jiawei Yao, Enbei Liu, Maham Rashid, and Juhua Hu. Augdmc: Data augmentation guided deep multiple clustering. *Procedia Computer Science*, 222:571–580, 2023.

Jiawei Yao, Qi Qian, and Juhua Hu. Customized multiple clustering via multi-modal subspace proxy learning. *Advances in Neural Information Processing Systems*, 37:82705–82725, 2024a.

Jiawei Yao, Qi Qian, and Juhua Hu. Multi-modal proxy learning towards personalized visual multiple clustering. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14066–14075. IEEE, 2024b.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11941–11952. IEEE, 2023.

Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European conference on computer vision*, pp. 649–666. Springer, 2016.

A SEMANTIC SELECTIVITY OF SVD-DERIVED BASIS VECTORS

To investigate the semantic selectivity of the basis vectors used in CSBR, we evaluate the discriminability of each basis vector across different semantic criteria. Given the LLM-generated text basis $\mathbf{T} \in \mathbb{R}^{m \times d}$ for a target criterion, we perform singular value decomposition:

$$\mathbf{T} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top. \tag{18}$$

The right singular vectors $\{\mathbf{v}_j\}_{j=1}^{r}$ form an orthonormal basis for the conditional subspace and are ordered by their corresponding singular values, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r$.

Given image embeddings $\{\mathbf{x}_n\}_{n=1}^{N}$, we compute the projection coefficient of the n-th image onto the j-th basis vector as:

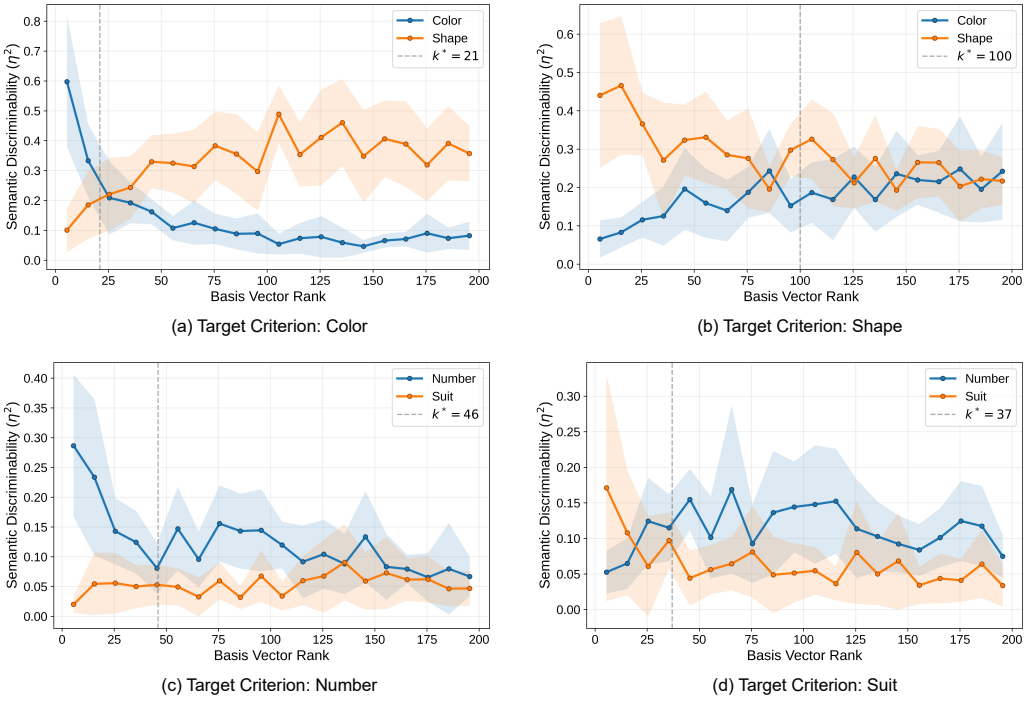
$$z_{n,j} = \mathbf{x}_n^\top \mathbf{v}_j. \tag{19}$$

For each semantic criterion C, we quantify the semantic discriminability of the j-th basis vector using the explained-variance ratio η^2:

$$\eta^2_{j,c} = \frac{\mathrm{Var}_{y^{(c)}}\left(\mathbb{E}[z_j \mid y^{(c)}]\right)}{\mathrm{Var}(z_j)}. \tag{20}$$

Here, $y^{(c)}$ denotes the labels under criterion C, and the variance over $y^{(c)}$ is weighted by the empirical class proportions. The ratio $\eta^2_{j,c} \in [0, 1]$ measures the proportion of variation along the j-th basis vector explained by criterion C, where a larger value indicates stronger semantic discriminability.

We conduct the analysis on two criterion pairs, Color-Shape (Clevr4-10k dataset) and Number-Suit (Cards dataset). For each pair, we alternately treat each criterion as the target criterion and construct its corresponding text basis.



(a) Target Criterion: Color

(b) Target Criterion: Shape

(c) Target Criterion: Number

(d) Target Criterion: Suit

Figure 3: Semantic discriminability of SVD-derived basis vectors across different criteria. Basis vectors are ranked by descending singular values. Solid curves represent the mean η^2 for each bin of 10 basis vectors, while shaded regions denote the corresponding standard deviation. Dashed vertical lines indicate the curvature-based truncation boundary k^* determined by CSBR.

As shown in Figure 3, basis vectors associated with larger singular values consistently exhibit higher semantic discriminability for the target criterion than for the non-target criterion. As the singular value decreases, the discriminability gap gradually narrows, indicating weaker semantic selectivity.

These results suggest that high-singular-value basis vectors are criterion-specific, capturing more criterion-relevant semantics. In contrast, low-singular-value basis vectors capture relatively more irrelevant semantics, increasing the risk of semantic leakage in the resulting conditional subspace. Notably, the curvature-based truncation boundary k^* generally lies near the region where the semantic selectivity of the basis vectors begins to diminish, supporting the effectiveness of our truncation strategy.

B Semantic Leakage Analysis via Linear Probing

To quantitatively assess the effectiveness of SP-CRL, we conduct a linear probing analysis on the Clevr4-10k and Cards datasets. Given a representation conditioned on the target criterion, we train separate linear probes to predict the labels of the corresponding images under the target and non-target criteria, respectively. Higher prediction accuracy on the target criterion indicates better preservation of target semantics, whereas lower prediction accuracy on the non-target criterion indicates more effective suppression of semantic leakage. On the Clevr4-10k dataset, we evaluate the Color and Shape criteria, using each criterion in turn as the target criterion and the other as the non-target criterion. On the Cards dataset, we evaluate the Number and Suit criteria in the same manner. For each setting, we randomly sample 1,000 images from the corresponding dataset for training and use all remaining images for testing. We repeat this procedure with five random seeds and report the average test accuracy.

As shown in Table 7, SP-CRL consistently achieves the lowest non-target prediction accuracy across all evaluated settings. On the Clevr4-10k dataset, when Color is the target criterion and Shape is the non-target criterion, SP-CRL reduces the non-target accuracy from 97.85% with orthogonal projection to 66.38%, while maintaining a high target accuracy of 97.52%. When their roles are reversed, the non-target accuracy decreases from 97.77% to 89.76%, while the target accuracy even improves slightly from 98.21% to 98.45%. Similar trends are observed on the Cards dataset. These results demonstrate that SP-CRL effectively suppresses semantic leakage while largely preserving target semantics.

Table 7: Linear probing prediction accuracy under target and non-target criteria.

Dataset	Target / Non-target	CLIP		Orthogonal Projection		SP-CRL	
		Target ↑	Non-target ↓	Target ↑	Non-target ↓	Target ↑	Non-target ↓
Cards	Number / Suit	**77.00**	72.82	64.31	60.85	62.61	**53.96**
	Suit / Number	**88.99**	56.19	84.16	42.02	82.89	**33.02**
Clevr4-10k	Color / Shape	**98.84**	98.98	98.68	97.85	97.52	**66.38**
	Shape / Color	**99.07**	98.68	98.21	97.77	98.45	**89.76**
Mean		**90.98**	81.67	86.34	74.62	85.37	**60.78**

C Datasets and Corresponding Prompts

In this section, we provide detailed descriptions of the datasets used in our experiments, along with the corresponding prompts for LLM and VLM. We organize the content by dataset, presenting the dataset structure and the specific prompts employed for each criterion.

C.1 Clevr4-10k

C.1.1 Dataset Structure

Clevr4-10k (Vaze et al., 2023) is a synthetic dataset built upon the CLEVR dataset. It consists of 10,531 images featuring multiple 3D objects positioned within fixed scenes. Each image in Clevr4-10k is annotated with all four criteria simultaneously, allowing for comprehensive evaluation of conditional representation learning methods. Table 8 shows the structure of the Clevr4-10k dataset.

Table 8: Structure of the Clevr4-10k dataset.

Criterion	Number of Classes	Class Examples
Shape	10	cone, cube, cylinder, diamond, gear, ...
Texture	10	brick, checkered, chessboard, circles, emojis, ...
Color	10	blue, brown, cyan, gray, green, ...
Count	10	1, 2, 3, 4, 5, ...

C.1.2 LLM PROMPTS

For Clevr4-10k, we design criterion-specific prompts to query the LLM for generating descriptive texts. Below we present the prompts used for each criterion:

```
Please generate common expressions to describe the shape (Texture / Color
/ Count), as many as possible, formatted as:  ["...", "...", "..."].
Ensure all items are unique and written in a single line, without any
nested lists or additional formatting.  Only generate the list, and do
not include any additional information.
```

C.1.3 VLM PROMPTS

For encoding the LLM-generated texts using the VLM, we construct prompts that combine the criterion with each descriptive text. The general template is:

```
Objects with the [CRITERION] of [TEXT].
```

where `[CRITERION]` is replaced with the specific criterion (e.g., "shape", "texture", "color", "count") and `[TEXT]` is replaced with each generated descriptive text.

C.2 CARDS

C.2.1 DATASET STRUCTURE

The Cards dataset (Yao et al., 2023) is a poker card dataset comprising 8,029 images of playing cards. Each card is photographed under various conditions, providing natural variations in lighting, angle, and background. Table 9 shows the structure of the Cards dataset.

Table 9: Structure of the Cards dataset.

Criterion	Number of Classes	Class Examples
Number	13	Ace, Two, three, ..., Jack, Queen, King
Suit	4	Clubs(♣), Diamonds(♢), Hearts(♡), Spades(♠)

C.2.2 LLM PROMPTS

For Cards, we design criterion-specific prompts to query the LLM for generating descriptive texts. Below we present the prompts used for each criterion:

```
Please generate common expressions to describe the number (suit) of
playing cards, as many as possible, formatted as:  ["...", "...", "..."].
Ensure all items are unique and written in a single line, without any
nested lists or additional formatting.  Only generate the list, and do
not include any additional information.
```

C.2.3 VLM PROMPTS

For encoding the LLM-generated texts using the VLM, we construct prompts that combine the criterion with each descriptive text. The general template is:

```
A poker card with the [CRITERION] of [TEXT].
```

where `[CRITERION]` is replaced with the specific criterion (e.g., "number", "suit") and `[TEXT]` is replaced with each generated descriptive text.

C.3 FLOWERS

C.3.1 DATASET STRUCTURE

The Flowers dataset is a real-world flower dataset constructed in-house from the Flowers102 dataset (Nilsback & Zisserman, 2008). Table 10 shows the structure of the Flowers dataset.

Table 10: Structure of the Flowers dataset.

Criterion	Number of Classes	Class Examples
Species	4	daisy, hyacinth, lily, rose
Color	5	pink, purple, red, white, yellow

C.3.2 LLM PROMPTS

For Flowers, we design criterion-specific prompts to query the LLM for generating descriptive texts. Below we present the prompts used for each criterion:

```
Please generate common expressions to describe the species (color) of flowers, as many as possible, formatted as: ["...", "...", "..."]. Ensure all items are unique and written in a single line, without any nested lists or additional formatting. Only generate the list, and do not include any additional information.
```

C.3.3 VLM PROMPTS

For encoding the LLM-generated texts using the VLM, we construct prompts that combine the criterion with each descriptive text. The general template is:

```
Flowers with the [CRITERION] of [TEXT].
```

where `[CRITERION]` is replaced with the specific criterion (e.g., "species", "color") and `[TEXT]` is replaced with each generated descriptive text.

C.4 DEEPFASHION

C.4.1 DATASET STRUCTURE

DeepFashion (Liu et al., 2016) is a large-scale fashion dataset widely used for clothing-related computer vision tasks. Following previous works (Ma et al., 2020), we use the category and attribute prediction benchmark split, which consists of 221,000 training images, 27,000 validation images, and 27,000 test images. Each image in DeepFashion is annotated with multiple attribute criteria simultaneously. Table 11 shows the structure of the DeepFashion dataset.

Table 11: Structure of the DeepFashion dataset.

Criterion	Number of Classes	Class Examples
Texture	156	abstract, animal, baroque, camo, circle, ...
Fabric	218	acid, bead, canvas, cloud, feather, ...
Shape	180	a-line, boxy, cropped, flared, shirt, ...
Part	216	bell, bow, buttoned, cowl, flat, ...
Style	230	baseball, bike, eagle, fox, mickey, ...

C.4.2 LLM PROMPTS

For DeepFashion, we design criterion-specific prompts to query the LLM for generating descriptive texts. Below we present the prompts used for each criterion:

```
Please generate common expressions to describe the texture (fabric / shape / part / style) of fashion items, as many as possible, formatted as: ["...", "...", "..."]. Ensure all items are unique and written in a single line, without any nested lists or additional formatting. Only generate the list, and do not include any additional information.
```

C.4.3 VLM PROMPTS

For encoding the LLM-generated texts using the VLM, we construct prompts that combine the criterion with each descriptive text. The general template is:

```
A fashion with the [CRITERION] of [TEXT].
```

where `[CRITERION]` is replaced with the specific criterion (e.g., "texture", "fabric", "shape", "part", "style") and `[TEXT]` is replaced with each generated descriptive text.

D AUTOMATIC INFERENCE OF NON-TARGET CRITERIA

The noise subspace identification described in Section 3.3 assumes that non-target criteria are specified in advance. In practical scenarios, however, such criteria may not always be available. To address this issue, we provide an MLLM-based procedure to automatically infer non-target criteria for noise subspace identification.

Removal of Target Semantic Components. Given a target criterion C_t, we remove the components of the image embeddings that lie within the corresponding conditional subspace:

$$\mathbf{I}_{\text{res}} = \mathbf{I} - \mathbf{I}(\mathbf{T}_t^*)^\top \mathbf{T}_t^*, \tag{21}$$

where $\mathbf{I}$ denotes the image embeddings and $\mathbf{T}_t^*$ denotes the refined basis of the target subspace. This operation reduces the influence of target semantics on subsequent non-target criterion inference.

Clustering-Based Representative Selection. We then perform k-means clustering on the residual embeddings to reveal their semantic structures. The number of clusters K is set according to the semantic diversity of the dataset. Let $\mathbf{i}_i^{\text{res}}$ denote the residual embedding of the i-th image. The clustering objective is formulated as:

$$\min_{\{z_i\},\{\boldsymbol{\mu}_k\}} \sum_{i=1}^{N} \|\mathbf{i}_i^{\text{res}} - \boldsymbol{\mu}_{z_i}\|_2^2, \tag{22}$$

where z_i denotes the cluster assignment of the i-th image and $\boldsymbol{\mu}_k$ denotes the centroid of the k-th cluster.

To obtain representative images from the clusters, we select the image closest to each cluster centroid:

$$r_k = \arg\min_{i:z_i=k} \|\mathbf{i}_i^{\text{res}} - \boldsymbol{\mu}_k\|_2^2. \tag{23}$$

The resulting representative image set is denoted by:

$$\mathcal{X}_{\text{rep}} = \{x_{r_1}, x_{r_2}, \ldots, x_{r_K}\}. \tag{24}$$

Non-Target Criterion Inference. For non-target criterion inference, we design an MLLM prompt template P_3 to identify salient non-target criteria from the representative images:

```
Given representative images from different clusters of the same dataset
and the target criterion [TARGET CRITERION], please identify a small
number of salient visual criteria, other than the target criterion,
that explain the differences among these clusters.  Ensure all items are
unique and formatted as:  ["...", "...", "..."].  Only generate the list,
and do not include any additional information.
```

Finally, the non-target criterion set is inferred by the MLLM as:

$$\mathcal{C}_{\text{nt}} = \text{MLLM}(P_3, C_t, \mathcal{X}_{\text{rep}}). \tag{25}$$

The inferred non-target criteria can be used for noise subspace identification in NSDP.

E VISUALIZATION OF CONDITIONAL REPRESENTATIONS

Figure 4 presents t-SNE visualizations of learned representations on the Clevr4-10k dataset under the "Color" criterion. CLIP's general representations (a) exhibit severe entanglement across semantic categories, failing to capture criterion-specific distinctions. While CRL (b) shows improved cluster formation, considerable inter-cluster overlap persists. In contrast, SP-CRL (c) achieves well-separated and compact clusters, demonstrating superior discriminative performance.

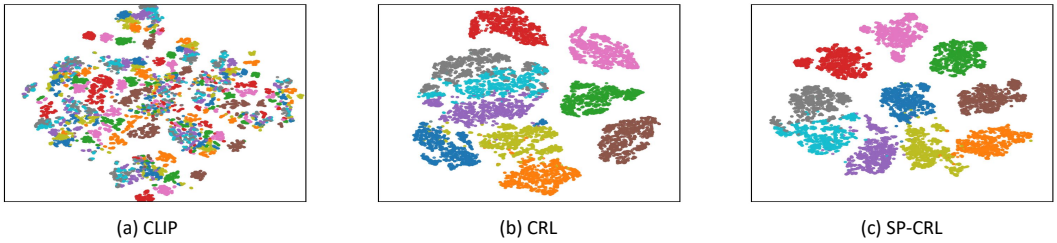


Figure 4: T-SNE visualizations of the representations learned by CLIP, CRL, and SP-CRL on the Clevr4-10k dataset for the "color" criterion.

F Sensitivity Analysis

F.1 Sensitivity Analysis on Basis Quantity.

In practical scenarios, the optimal number of text basis vectors remains unknown. To validate the robustness of SP-CRL to text basis quantity, we examine its performance on the customized clustering task under varying numbers of basis vectors. As shown in Figure 5, we conduct experiments on both Clevr4-10k and Cards datasets with the number of basis vectors ranging from 5 to 2000. SP-CRL exhibits stable and superior performance across all criteria, remaining insensitive to the number of basis vectors. In contrast, CRL demonstrates significant performance variations, particularly in color-based clustering on Clevr4-10k. Moreover, regardless of the initial basis cardinality, SP-CRL utilizes only a small set of refined basis vectors constructed through Criterion-Specific Basis Refinement. This enables SP-CRL to achieve significant efficiency in scenarios where the semantics corresponding to the criterion are overly broad.

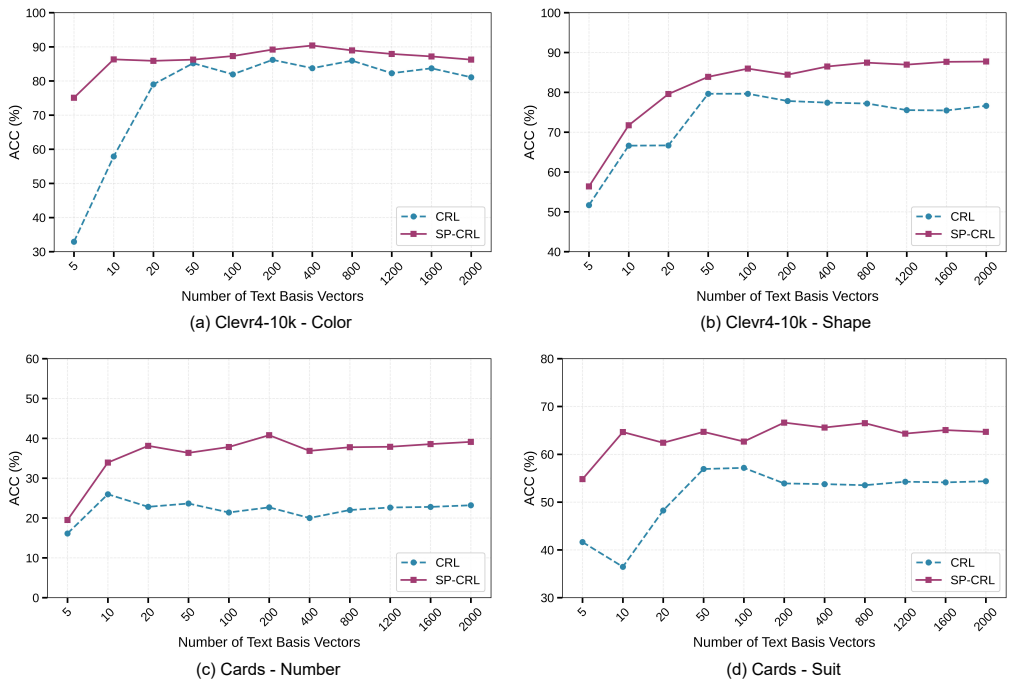


Figure 5: Performance with different numbers of basis vectors.

F.2 Sensitivity Analysis on LLM Selection.

SP-CRL leverages criterion-specific descriptions generated by LLMs to construct text bases. By default, we adopt GPT-4 (Achiam et al., 2023) as the LLM for description generation. To validate the robustness of SP-CRL to LLM selection, we evaluate its performance on customized clustering across a variety of LLMs, including

Claude 4, Gemini 2.5 (Comanici et al., 2025), and DeepSeek-v3 (Liu et al., 2024a). As shown in Table 12, experiments are conducted on the Clevr4-10k and Cards datasets, covering the Shape, Color, Number, and Suit criteria. Under different LLM selections, SP-CRL maintains stable performance. The average standard deviation across all metrics is only 0.89%, demonstrating the robustness of SP-CRL to the choice of description generator.

Table 12: Performance under Different LLM Selections.

LLM	Clevr4-10k						Cards					
	Shape			Color			Number			Suit		
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI
GPT-4	86.67	90.69	84.11	89.99	89.88	85.09	38.78	44.45	28.40	41.61	73.22	42.20
Claude 4	86.54	88.94	83.36	89.73	90.94	85.11	38.45	42.79	27.54	40.60	72.56	41.26
Gemini 2.5	85.29	88.49	82.12	88.71	88.65	82.86	37.66	40.42	25.70	41.58	71.68	41.85
DeepSeek-v3	85.14	87.57	81.29	88.67	87.65	82.32	38.07	42.37	26.37	40.15	71.96	40.70

F.3 Sensitivity Analysis on VLM Selection.

SP-CRL relies on vision-language models to encode both image embeddings and text bases. To evaluate the robustness of SP-CRL to VLM selection, we conduct customized clustering experiments using CLIP (Radford et al., 2021), BLIP2 (Li et al., 2023a), and SigLIP (Zhai et al., 2023) as the underlying models. As shown in Table 13, experiments are conducted on the Clevr4-10k and Cards datasets, covering the Shape, Color, Number, and Suit criteria. Under different VLM selections, SP-CRL consistently outperforms both the original VLMs and CRL on most evaluation metrics. For example, on Clevr4-10k, CLIP+SP-CRL improves the Shape NMI from 78.40% to 86.67% over CLIP+CRL, while on Cards, BLIP2+SP-CRL improves the Number ACC from 48.55% to 61.90%. These results demonstrate the strong generalization of SP-CRL across different VLM backbones.

Table 13: Performance under Different VLM Selections.

Method	Clevr4-10k						Cards					
	Shape			Color			Number			Suit		
	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI
CLIP	74.22	73.19	64.15	0.83	12.23	0.27	16.84	18.91	8.44	16.52	43.74	12.93
CLIP+CRL	78.40	81.83	71.35	88.09	87.26	81.07	24.68	28.59	12.27	37.70	62.08	34.36
CLIP+SP-CRL	**86.67**	**90.69**	**84.11**	**89.99**	**89.88**	**85.09**	**38.78**	**44.45**	**28.40**	**41.61**	**73.22**	**42.20**
BLIP2	86.98	85.68	81.17	0.99	11.92	0.24	24.34	25.25	13.08	31.26	47.04	22.25
BLIP2+CRL	90.01	87.97	83.35	83.44	81.78	74.37	46.40	48.55	33.47	59.43	71.96	53.00
BLIP2+SP-CRL	**93.20**	**93.01**	**90.26**	**88.86**	**90.39**	**83.68**	**58.07**	**61.90**	**48.76**	**62.52**	**83.33**	**62.71**
SigLIP	89.75	88.24	84.51	0.41	11.61	0.08	36.89	34.51	22.79	38.19	51.23	29.00
SigLIP+CRL	91.56	**91.64**	87.78	83.39	80.19	72.83	78.43	72.75	63.77	71.88	75.95	65.38
SigLIP+SP-CRL	**93.43**	90.81	**87.96**	**88.78**	**85.62**	**81.30**	**87.52**	**89.12**	**84.56**	**80.36**	**93.86**	**84.03**