
CF-VLM: Counterfactual Vision-Language Fine-tuning

(More detailed supplement)

Anonymous Author(s)

Affiliation

Address

email

Abstract

Recent advances in vision-language models (VLMs) have greatly improved cross-modal semantic understanding, yet significant limitations remain in fine-grained discrimination and deep causal reasoning tasks. Existing VLMs often rely on superficial statistical correlations, lacking the ability to capture the underlying causal logic between visual and textual content. To address this, we propose CounterFactual Vision-Language Fine-tuning (CF-VLM), a novel framework that enhances the causal reasoning capabilities of VLMs through the targeted use of counterfactual samples. CF-VLM introduces three complementary training objectives: maintaining foundational cross-modal alignment, reinforcing the uniqueness, and stability of factual scene representations against coherent counterfactuals, and sharpening the model’s sensitivity to minimal but critical causal edits. Extensive experiments demonstrate that CF-VLM consistently outperforms strong baselines and state-of-the-art methods on compositional reasoning and generalization benchmarks. Furthermore, it shows promise in mitigating visual hallucinations, indicating improved factual consistency. Our CF-VLM provides a robust foundation for deploying VLMs in high-stakes, real-world scenarios requiring reliable reasoning and interpretability.

1 Introduction

With the rapid advancement of vision-language models (VLMs)[1–5] such as CLIP [6], the capability of cross-modal semantic understanding has significantly improved, laying a solid foundation for diverse artificial intelligence applications [1]. However, current VLMs still exhibit fundamental limitations in complex visual-language tasks that demand fine-grained discrimination and deep reasoning[7–11]. These models tend to rely on superficial statistical correlations, struggling to capture the underlying causal logic between visual and textual content[4, 5, 12]. Such limitations are particularly evident in fine-grained recognition and causal reasoning scenarios, constituting a core bottleneck that hampers the applicability of VLMs in high-stakes, real-world environments requiring robust reasoning[13].

To improve fine-grained alignment, recent efforts have extended contrastive learning strategies[14–16], commonly optimizing objectives like the triplet loss $L = \sum [m + S(A, N) - S(A, P)]_+$, where $S(\cdot, \cdot)$ denotes similarity and m is a margin. These methods aim to enforce $S(A, P) > S(A, N) + m$ in the representation space [17], improving discriminative performance. Nevertheless, their primary focus lies in binary distinction—whether pairs match—rather than in understanding the causal attributes that explain how minimal differences in counterfactuals may lead to a semantic shift[5, 18, 19]. Consequently, these models often fail to attend to the decisive attributes or relations that govern semantic matching, leading to misinterpretation when faced with counterfactual perturbations.

To overcome these limitations, we propose **CF-VLM** (CounterFactual Vision-Language Fine-tuning), a novel fine-tuning framework designed to enhance the causal reasoning capability of VLMs. Unlike

prior approaches focused solely on improving general discrimination, CF-VLM centers the learning process around *counterfactual samples* [20]. Specifically, *counterfactual samples* are crafted by applying minimal yet semantically crucial edits to an original image or its associated causal relations. These edits alter the logical meaning of the scene while preserving proximity to the factual base. For instance, changing the color, category, status, or spatial configuration of a core object, or reversing an action-outcome relation, can generate a new counterfactual scenario. Though seemingly subtle, these modifications result in fundamental changes in the semantic match with the accompanying text. CF-VLM thus encourages the model to detect and reason about such *causal decision points*—the critical attributes or relations that determine whether a visual-textual pair truly matches. Building upon cross-modal alignment, CF-VLM introduces two progressive and complementary training objectives to model the contrastive relationships between factual and counterfactual data:

First, our CF-VLM reinforces the uniqueness and stability of the *anchor factual scene* ($I_{\text{anchor}}, T_{\text{anchor}}$) by contrasting it with a set of *complete counterfactual scenarios* ($I_{\text{cf}_i}, T_{\text{cf}_i}$), where both image and text are jointly modified to represent a logically consistent yet semantically distinct alternative. The training objective enforces $S(I_{\text{anchor}}, T_{\text{anchor}}) \gg S(I_{\text{cf}_i}, T_{\text{cf}_i})$ for all i , thus establishing a clear reference and preventing representation confusion caused by the presence of numerous “parallel realities”.

Second and more critically, our CF-VLM focuses on enhancing the model’s sensitivity to *minimal causal edits*—those that singularly affect the semantic validity of a factual description. In this setting, the factual image I_{fact} is paired with its original text T_{fact} (with similarity $S(I_{\text{fact}}, T_{\text{fact}})$), while the same text is also paired with a minimally edited image I_{cf} (with similarity $S(I_{\text{cf}}, T_{\text{fact}})$). CF-VLM aims to maximize the margin $S(I_{\text{fact}}, T_{\text{fact}}) \gg S(I_{\text{cf}}, T_{\text{fact}})$, thereby guiding the model to focus on the semantic shift caused by critical causal modifications.

2 Preliminary

Before detailing the CF-VLM framework, this section introduces foundational concepts, key notations, and the standard formulation of vision-language models (VLMs) in cross-modal semantic understanding. Representation Learning in Vision-Language Models: The core objective of VLMs is to learn joint multi-modal embeddings that align image and text representations in a shared semantic space[21–23]. Given an image I and its corresponding textual description T , a typical VLM consists of two primary encoder components: **Image Encoder** $f_I(\cdot)$: This module maps the input image I to a d -dimensional visual embedding vector $\mathbf{e}_I \in \mathbb{R}^d$: $\mathbf{e}_I = f_I(I)$ **Text Encoder** $f_T(\cdot)$: Analogously, the text encoder transforms the input textual sequence T into a d -dimensional text embedding vector $\mathbf{e}_T \in \mathbb{R}^d$: $\mathbf{e}_T = f_T(T)$. To ensure the comparability of these embeddings in the metric space and to stabilize similarity computations, both \mathbf{e}_I and \mathbf{e}_T are typically L_2 -normalized[24] before being used in downstream contrastive learning objectives, resulting in unit-length representations.

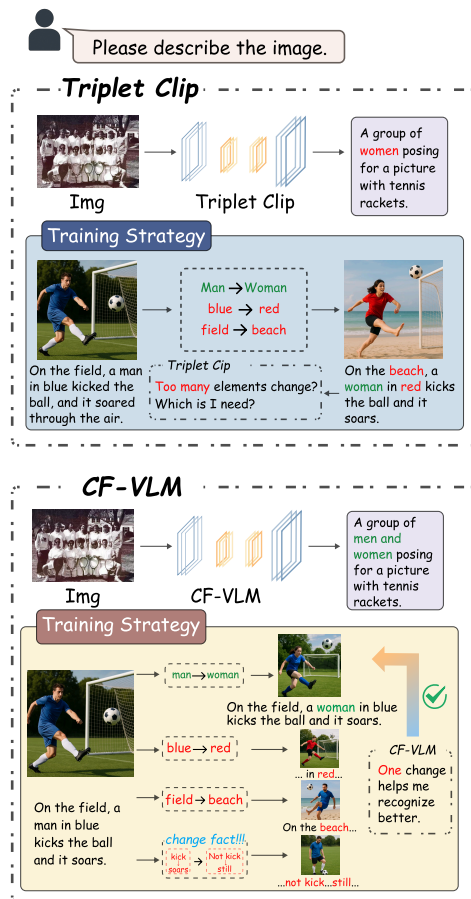


Figure 1: Illustration of CF-VLM’s training framework. The model is exposed to both factual and counterfactual image-text pairs, where the latter introduce minimal but semantically decisive edits (e.g., “kick” \rightarrow “not kick”, “man” \rightarrow “woman”). In contrast to Triplet CLIP, which focuses on coarse-grained similarity (e.g., matching vs. non-matching), CF-VLM targets fine-grained causal decision points, guiding the model to recognize critical semantic shifts.

93 2.1 Cross-Modal Similarity Measurement

94 The learned image embedding \mathbf{e}_I and text embedding \mathbf{e}_T reside in a shared representation space[25].
 95 Within this space, semantically aligned image-text pairs are expected to exhibit high similarity,
 96 whereas unrelated pairs should demonstrate low similarity. This cross-modal semantic similarity,
 97 denoted as $S(I, T)$, is typically measured via the cosine similarity between the normalized embedding
 98 vectors, which is mathematically equivalent to their dot product: $S(I, T) = \mathbf{e}_I^\top \mathbf{e}_T = \frac{f_I(I) \cdot f_T(T)}{\|f_I(I)\| \cdot \|f_T(T)\|}$.
 99 In practice, since both embedding vectors are L2-normalized[24], the above formulation simplifies to:
 100 $S(I, T) = \mathbf{e}_I^\top \mathbf{e}_T$. This similarity score serves as a fundamental computational unit in the contrastive
 101 learning objective.

102 2.2 Definition and Types of Counterfactual Samples

103 Our CF-VLM leverages counterfactual samples for fine-grained supervision. While most existing
 104 vision-language learning methods rely on general data augmentation or negative sampling [26], such
 105 approaches often lack the capacity to systematically generate the minimal yet semantically critical
 106 interventions required for deep causal probing. In contrast, we define a *counterfactual sample* as
 107 a new image-text pair, derived from an anchor (I_a, T_a) through a precise, targeted intervention.
 108 These edits—guided by prompt engineering for controlled and interpretable manipulation—are
 109 specifically designed to enhance the model’s understanding of key causal factors underlying image-
 110 text alignment. This focus distinguishes our CF-VLM from those methods that merely increase data
 111 diversity or construct generic hard negatives. We concentrate on two principal types of counterfactual
 112 interventions: (1) **Single Object Attribute Modification**: Targeting a core object in the image,
 113 this intervention modifies only one physical attribute at a time—such as color (e.g., red apple →
 114 green apple), category (dog → cat), pose or state (standing → lying), spatial relation (left → right),
 115 or quantity. The objective is to rigorously test and enhance the model’s sensitivity to subtle yet
 116 semantically disruptive changes [27]. (2) **Key Causal Relationship Adjustment**: This intervention
 117 modifies the key action, event outcome, or causal chain depicted in the scene (e.g., changing “hits the
 118 ball” to “misses the ball,” or “object falling” to “object floating”), directly challenging the model’s
 119 understanding of visual dynamics and physical plausibility. Based on these targeted intervention
 120 strategies, we construct two categories of counterfactual data structures for training:

- 121 • **Complete Counterfactual Scenario Pair** (I_{cfk}, T_{cfk}) : Both the image and the text are
 122 jointly edited to form a logically coherent but factually distinct scenario from the anchor,
 123 facilitating the learning of plausible alternative realities.
- 124 • **Minimally Edited Counterfactual Image** $I_{cf_edit_j}$: A single critical intervention is applied
 125 to I_a , and the resulting $I_{cf_edit_j}$ is paired with the original anchor text T_a as a hard negative
 126 sample, enabling fine-grained discrimination.

127 3 Methodology

128 **CF-VLM Overview**: Our CF-VLM is built upon a pretrained vision-language model and aims to
 129 refine its representational space by introducing carefully constructed counterfactual samples along
 130 with a set of targeted learning objectives[28–30]. The central insight of CF-VLM is that merely
 131 learning to distinguish between matching and non-matching image-text pairs is insufficient. More
 132 crucially, the model must develop an understanding of *why* a given pair matches or not—particularly
 133 when the differences are subtle and involve causally significant attributes[31]. To this end, the overall
 134 training objective of CF-VLM, denoted as L_{CF-VLM} , comprises three key loss components, each
 135 responsible for a distinct but complementary aspect of model enhancement: Maintaining the model’s
 136 foundational cross-modal alignment capability via L_{align} ; Strengthening the uniqueness and stability
 137 of anchor factual scene representations relative to plausible counterfactuals via L_{csd} ; Enhancing the
 138 model’s causal sensitivity to semantic changes induced by minimal but critical edits via L_{fcd} . These
 139 components are integrated through a weighted sum: $L_{CF-VLM} = \alpha \cdot L_{align} + \beta \cdot L_{csd} + \gamma \cdot L_{fcd}$ where
 140 α , β , and γ are hyperparameters that balance the contribution of each loss.

141 3.1 Counterfactual Data for Training

142 The organization of training data plays a pivotal role in fulfilling the learning objectives of CF-VLM.
 143 For each training iteration, our CF-VLM is provided with a structured data unit comprising the

following elements: An anchor factual image-text pair (I_a, T_a) ; a set of K complete counterfactual scenario pairs $(I_{cf_k}, T_{cf_k})_{k=1}^K$, where each (I_{cf_k}, T_{cf_k}) is derived from a targeted modification of (I_a, T_a) and collectively forms a new, internally coherent counterfactual scene; and a set of J minimally edited counterfactual images $I_{cf_edit_j} j = 1^J$, where each $I_{cf_edit_j}$ is generated by applying a single critical semantic edit—such as modifying an object attribute or altering a causal relation—to the anchor image I_a . This counterfactual data structure enables CF-VLM to perform targeted supervision across multiple levels of semantic granularity[32, 33], ranging from basic alignment to causal sensitivity.

3.2 Core Learning Objectives of CF-VLM

Foundational Cross-Modal Alignment Loss. To ensure that CF-VLM preserves the foundational image-text alignment capability of the pretrained vision-language model during fine-grained tuning, we incorporate a standard contrastive loss. This loss is consistent with the pretraining objectives used in models such as CLIP and aims to pull semantically relevant image-text pairs closer in the shared embedding space, while pushing apart unrelated pairs. Given a training batch of N factual image-text pairs $\{(I_i, T_i)\}_{i=1}^N$, let \mathbf{e}_{I_i} and \mathbf{e}_{T_j} denote the normalized embeddings of image I_i and text T_j , respectively. Their similarity score is computed as: $S(I_i, T_j) = \mathbf{e}_{I_i}^\top \mathbf{e}_{T_j}$. The foundational alignment loss L_{align} is defined using a symmetric InfoNCE objective:

$$L_{\text{align}} = -\frac{1}{2N} \sum_{i=1}^N \left(\log \frac{\exp(S(I_i, T_i)/\tau)}{\sum_{j=1}^N \exp(S(I_i, T_j)/\tau)} + \log \frac{\exp(S(I_i, T_i)/\tau)}{\sum_{j=1}^N \exp(S(I_j, T_i)/\tau)} \right) \quad (1)$$

Here, $S(I_i, T_j)$ denotes the cosine similarity between the normalized image embedding \mathbf{e}_{I_i} and text embedding \mathbf{e}_{T_j} . The first term inside the sum, $-\log \frac{\exp(S(I_i, T_i)/\tau)}{\sum_{j=1}^N \exp(S(I_i, T_j)/\tau)}$, encourages each image I_i to be most similar to its paired text T_i , compared to all other texts in the batch. The second term, $-\log \frac{\exp(S(I_i, T_i)/\tau)}{\sum_{j=1}^N \exp(S(I_j, T_i)/\tau)}$, encourages each text T_i to be most similar to its paired image I_i , compared to all other images in the batch. The temperature parameter τ controls the sharpness of the distribution. This symmetric formulation ensures that both image-to-text and text-to-image alignments are jointly optimized.

Counterfactual Scenario Discrimination Loss. After establishing basic image-text alignment, CF-VLM introduces the loss term L_{csd} to reinforce the uniqueness and stability of the representation of the anchor factual scenario. The core idea is that the model should not only understand the current factual instance but also clearly distinguish it from logically coherent yet semantically different *parallel* counterfactual scenarios.

Given an anchor image-text pair (I_a, T_a) and a set of K complete counterfactual scenario pairs $\{(I_{cf_k}, T_{cf_k})\}_{k=1}^K$ derived from it, the L_{csd} loss enforces that the similarity score $S(I_a, T_a)$ of the anchor pair is significantly higher than that of any counterfactual pair $S(I_{cf_k}, T_{cf_k})$. The loss is implemented as a hinge loss:

$$L_{\text{csd}} = \frac{1}{K} \sum_{k=1}^K \max(0, S(I_{cf_k}, T_{cf_k}) - S(I_a, T_a) + m_1) \quad (2)$$

Here, $S(I_a, T_a)$ denotes the similarity between the anchor image-text pair, while $S(I_{cf_k}, T_{cf_k})$ represents the similarity of the k -th counterfactual pair derived from the anchor. For each counterfactual scenario, the loss encourages the similarity of the anchor pair to exceed that of the counterfactual by at least a margin m_1 . If the counterfactual pair is too close to or even exceeds the anchor in similarity, the hinge loss becomes positive and penalizes the model. When the anchor similarity already surpasses all counterfactuals by at least m_1 , the loss is zero. This mechanism enforces a clear separation between the factual anchor and its plausible but semantically different counterfactuals in the embedding space, thereby improving the model’s discriminative power. **Fine-Grained Causal Discrimination Loss.** The loss term L_{fcd} represents the core innovation of the CF-VLM framework with respect to causal understanding. It is specifically designed to enhance the model’s ability to detect and reason about changes in image-text matching relationships that arise from *minimal but semantically critical edits*, such as single attribute modifications or causal relationship reversals, as defined in Section 2.3. Given an anchor factual image-text pair (I_a, T_a) and a set of J **minimally edited counterfactual images** $\{I_{cf_edit_j}\}_{j=1}^J$ —each generated via an atomic semantic intervention on I_a —the objective of L_{fcd} is to ensure that the similarity score $S(I_a, T_a)$ is significantly higher

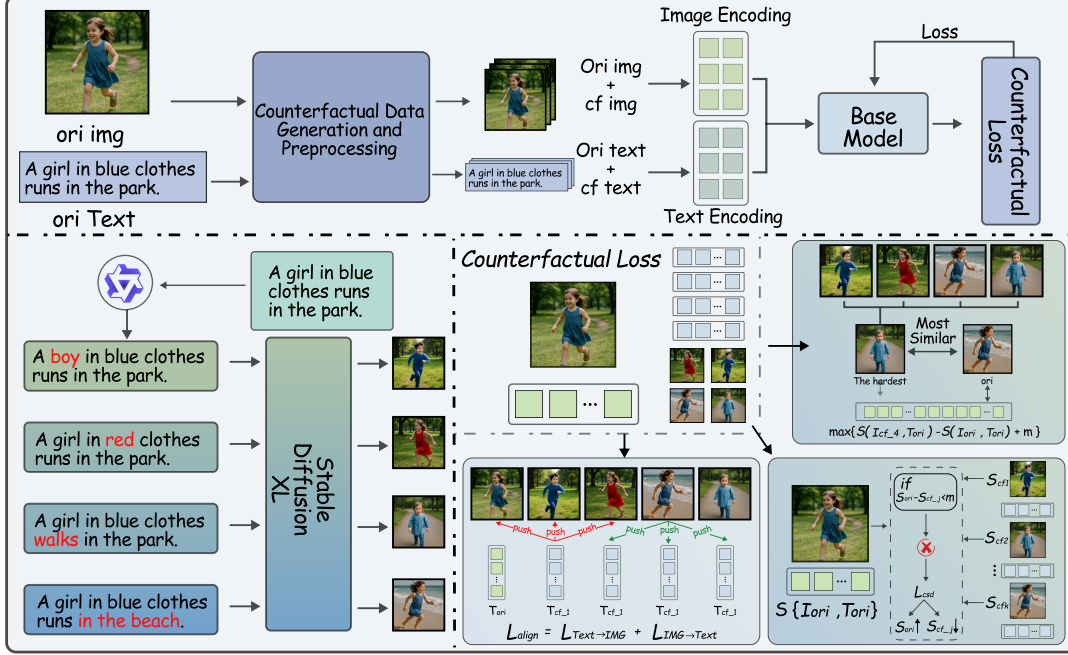


Figure 2: CF-VLM training pipeline. Given a factual image-text anchor, the framework generates complete and minimally edited counterfactual images using a fine-tuned SDXL model. The model is optimized via three complementary objectives: cross-modal alignment (L_{align}), counterfactual scenario discrimination (L_{csd}), and fine-grained causal discrimination (L_{fcd})—enhancing semantic precision and causal sensitivity.

than that between T_a and any $I_{cf_edit_j}$. To strengthen training effectiveness, we focus on the most confusing counterfactual (i.e., the one with the highest similarity to T_a), thereby forming a *hard negative* sample. The loss adopts a hinge formulation:

$$L_{fcd} = \max \left(0, \max_{j \in \{1..J\}} S(I_{cf_edit_j}, T_a) - S(I_a, T_a) + m_2 \right) \quad (3)$$

Here, $S(I_a, T_a)$ is the similarity between the anchor image and its paired text, while $S(I_{cf_edit_j}, T_a)$ denotes the similarity between the anchor text and the j -th minimally edited counterfactual image. Among all counterfactual images, only the one with the highest similarity to T_a (i.e., the hardest negative) is considered in the loss calculation. The hinge loss encourages $S(I_a, T_a)$ to exceed this hardest negative similarity by at least a margin m_2 ; otherwise, the loss penalizes the model. If all counterfactuals are sufficiently less similar than the anchor, the loss is zero. This mechanism ensures the model can robustly detect and distinguish subtle but semantically important changes introduced by minimal edits, thereby enhancing its fine-grained causal reasoning capabilities.

4 Experiment

We systematically evaluate our CF-VLM for vision-language models, i.e., Qwen-VL (7B)[34] and CLIP-ViT-B/32, with a focus on enhancing complex visual reasoning. The experimental setup covers pretraining and fine-tuning on filtered CC12M [35] (8.6M image-text pairs) and the CC3M [36] subset (2.6M pairs), with MSCOCO Captions [37] (120K pairs) optionally included for analysis. For each image-text pair, one counterfactual sample is generated by our dynamic counterfactual generation (DCF) strategy, effectively doubling the training data. **Evaluation is conducted on compositional and generalization benchmarks**, including the Conne [38], ARO [39], VL-Checklist [40], ImageNet-1k [41] (zero-shot classification), and MSCOCO/Flickr30k [42] (zero-shot retrieval). Comparisons include *Zero-shot Qwen-VL (7B)*, *Standard Fine-tuning*, *Text-Negative Fine-tuning*, leading *CLIP-based* models (TripletCLIP [16], CE-CLIP+ [43], COGT-CLIP [44], NegCLIP, Structure-CLIP [45]), *LLM-based VLMs* (LLaVA-1.5 [46], InstructBLIP [47], MiniGPT-4 [48]), and our CF-VLM (Qwen-VL fine-tuned on CC12M+DCF or CC3M+DCF). The counterfactual texts are generated by Qwen2-72B-Instruct with 3-shot chain-of-thought prompting, and images

Table 1: Performance of CLIP-based SOTA vision-language models.

Method	Replace-Obj	Replace-Attr	Replace-Rel	Conne (Avg)	VG-Rel	VG-Attr	ARO (Avg)	Attribute	Object	Relation	VL-Checklist (Avg)
NegCLIP (ViT-B/32, CC12M)	56.9	56.7	52.2	55.27	80.9	69.9	75.4	68.3	79.2	62.8	70.1
TripletCLIP (ViT-B/32, CC12M)	58.2	58.2	53.2	56.53	82.6	77.4	80	77.9	86.2	83.2	82.4
CE-CLIP+ (ViT-L/14, CC3M+VG)	57.5	57.9	53.8	56.4	83.6	76.2	79.9	76.3	85.2	75.3	78.9
Structure-CLIP (ViT-B/32, CC12M)	58.1	58.1	53.8	56.67	84.9	83.2	84.05	80.2	86.3	82.4	83.0
COGT-CLIP (ViT-B/32, CC12M)	58.8	60.7	54.2	57.9	87.3	88.2	87.75	84.2	81.4	86.4	84.0
Zero-shot (ViT-B/32)	50.9	51.7	46.6	49.73	59.7	61.9	60.8	68.2	81.9	63.9	71.3
Standard FT (ViT-B/32, CC12M)	55.7	56.4	51.2	54.43	74.2	68.2	71.2	71.4	83.1	69.4	74.6
CF-VLM (Ours, ViT-B/32, CC12M)	60.4	60.4	56.6	59.13	88.4	90.3	89.35	88.4	87.6	89.3	88.4

Table 2: Performance of LLM-based VLMs, including Qwen-VL and LLaVA variants.

Method	Replace-Obj	Replace-Attr	Replace-Rel	Conne (Avg)	VG-Rel	VG-Attr	ARO (Avg)	Attribute	Object	Relation	VL-Checklist (Avg)
InstructBLIP (FlanT5XL, Reported)	64.1	65.2	59.9	63.07	70.5	87.4	78.95	57.2	81.3	62.4	66.97
MiniGPT-4 (7B Vicuna, Reported)	74.3	73.2	76.7	74.73	48.2	57.4	52.8	71.4	83.6	86.4	80.47
Zero-shot Qwen-VL (7B)	80.7	78.4	79.4	79.5	78.3	81.4	79.85	85.6	86.4	87.9	86.63
Standard FT (Qwen-VL, CC12M)	83.4	82.1	82.3	82.6	86.3	87.2	86.75	87.4	87.9	88.3	87.87
TextNeg FT (Qwen-VL, CC12M)	84.6	84.2	83.6	84.13	87.5	87.5	87.5	87.9	88.3	89.1	88.43
CF-VLM (Ours, Qwen-VL 7B, CC12M)	88.3	87.5	86.9	87.57	91.8	94.6	93.2	89.6	90.7	91.4	90.57
LLaVA-1.5 (13B, Reported)	57.6	62.4	58.6	59.53	65.4	73.2	69.3	64.3	80.6	70.2	71.7
Standard FT (LLaVA-1.5, CC12M)	60.8	64.1	60.4	61.77	68.4	77.4	72.9	66.2	82.4	72.3	73.63
TextNeg FT (LLaVA-1.5, CC12M)	61.5	65.2	62.3	63.0	69.9	78.6	74.25	67.3	83.6	74.2	75.03
CF-VLM (Ours, LLaVA-1.5 7B, CC12M)	64.4	67.8	63.7	65.3	72.3	80.3	76.3	68.6	85.1	75.6	76.43

use SDXL 1.0 Base [49] + Refiner (40+15 step schedule). Models are trained using AdamW ($\beta_1 = 0.9, \beta_2 = 0.98, \epsilon = 1 \times 10^{-6}$), a peak learning rate of 1×10^{-5} (cosine decay, 500 warmup steps), weight decay 0.1, and batch size 256 on a single NVIDIA A100 (bf16). Training runs for 200K steps (CC12M) or 90K steps (CC3M), with each batch containing a 1:1 ratio of factual and counterfactual samples. The loss is a symmetric triplet contrastive loss $\mathcal{L} = \mathcal{L}_{I \rightarrow T} + \mathcal{L}_{T \rightarrow I}$ with a learnable temperature (init 0.07). **We report main results, ablations, and analyses** to show how counterfactual data aids hallucination mitigation and demonstrate CF-VLM’s advantages over SOTA methods. **For evaluation**, compositional reasoning benchmarks use accuracy; ImageNet-1k zero-shot classification is measured by top-1 and top-5 accuracy; image-text retrieval (MSCOCO, Flickr30k) is assessed via Recall@1/5/10. All results are averaged over three random seeds, with standard deviations reported.

4.1 Main Results and Analyses

Compositional Reasoning Performance. We conduct a comprehensive evaluation of CF-VLM on key compositional reasoning benchmarks, including CONNE, ARO, and VL-CHECKLIST, and compare its performance against both baseline models and relevant state-of-the-art (SOTA) approaches. The results are summarized in Table 1 and Table 2. **Performance Gains in CLIP-based Models.** As shown in Table 1, CF-VLM (ViT-B/32, CC12M) outperforms all CLIP-based baselines. On Conne, it achieves 59.13% accuracy, exceeding Std FT by +4.7 points and surpassing TripletCLIP/CE-CLIP+ by +2.6/+2.7 points. On Visual Genome, it obtains 88.4% (VG-Rel) and 90.3% (VG-Attr), outperforming previous models. CF-VLM also achieves 88.4% on the VL-Checklist, showing strong robustness across benchmarks. **Transferability and Advantage in LLM-based Architectures.** As shown in Table 2, CF-VLM delivers strong results with LLM-based models. For Qwen-VL 7B, it achieves 87.57% on Conne, improving over Std FT by +4.9 and over TextNeg FT by +3.4 points. On VG, it reaches 91.8% (Rel) and 94.6% (Attr). CF-VLM also transfers well to LLaVA-1.5, consistently outperforming Std FT and TextNeg FT, and surpasses strong instruction-tuned models like InstructBLIP and MiniGPT-4 across most tasks, demonstrating broad generalization and adaptability.

Generalization Performance on Standard Vision-Language Tasks. As shown in Figure 3, CF-VLM achieves strong generalization on standard tasks, including zero-shot ImageNet-1k classification (top-1: 8.83%, top-5: 19.54%) and cross-modal retrieval (R@5: 13.80% on MSCOCO(Avg), 26.58% on Flickr30k(Avg)), outperforming or matching CLIP-based SOTA and relevant baselines.

4.2 Ablation Studies

Effectiveness of Counterfactual Sample Generation. To assess the impact of counterfactual batches on multi-modal causal reasoning, we design three ablation baselines: (1) **Non-causal Baseline:** training with original pairs and random negatives; (2) **Text-only Counterfactual:** counterfactual

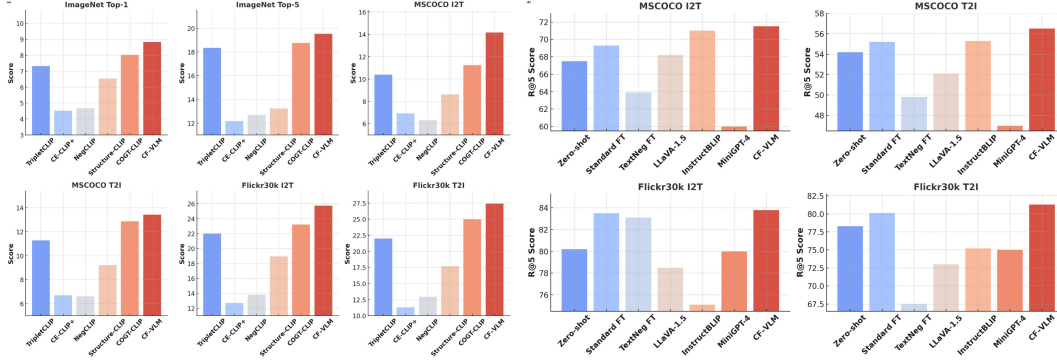


Figure 3: Generalization results. Left (3 columns): CF-VLM boosts baseline performance on CLIP-based classification and retrieval. Right (2 columns): LLM-based evaluation shows improved semantic alignment and Recall@5 over standard finetuning.

Table 3: Performance comparison of different ablation groups on causal reasoning benchmarks. CF-VLM consistently outperforms all baselines across tasks.

Model	ConMe (Avg)	Replace-Obj	Replace-Attr	Replace-Rel	ARO (Avg)	VG-Rel	VG-Attr
Non-causal Baseline	54.4	55.7	56.4	51.2	71.2	74.2	68.2
Text-only Counterfactuals	55.9	57.3	57.6	52.8	78.65	81.9	75.4
Image-only Counterfactuals	56.1	57.1	57.9	53.4	78.25	82.2	74.3
CF-VLM	59.1	60.4	60.4	56.6	89.35	88.4	90.3

251 texts with original images; (3) **Image-only Counterfactual**: counterfactual images with original
 252 texts. All experiments use the CF-VLM framework with consistent settings. As shown in Table 3,
 253 the full CF-VLM model with both image and text counterfactuals achieves the best results on all
 benchmarks (e.g., ConMe: 59.1, ARO: 89.35), outperforming all baselines.

Table 4: Ablation study of different loss components on the ConMe dataset. The full model with all three loss terms achieves the best average performance.

Foundational Cross-Modal Alignment Loss	Counterfactual Scenario Discrimination Loss	Fine-Grained Causal Discrimination Loss	Replace-Obj	Replace-Att	Replace-Rel	Avg
✓	✓	×	59.4	59.2	55.1	57.9
×	✓	✓	58.9	58.1	54.8	57.3
✓	×	✓	59.3	59.4	55.4	58.0
✓	×	×	57.4	57.8	53.5	56.2
×	✓	×	56.8	56.8	52.9	55.5
×	×	✓	56.6	56.4	53.1	55.4
✓	✓	✓	60.4	60.4	56.6	59.1

254

255 Effectiveness of Counterfactual Causal CLIP Losses

256 To evaluate the impact of different loss components, we perform an ablation study on ConMe using:
 257 (1) **Foundational Cross-Modal Alignment Loss**, (2) **Counterfactual Scenario Discrimination Loss**, and (3) **Fine-Grained Causal Discrimination Loss**. We test various configurations by adding
 258 or omitting each loss. The full model with all three losses achieves the best average accuracy (59.1),
 259 with 60.4 on *Replace-Obj*, 60.4 on *Replace-Attr*, and 56.6 on *Replace-Rel*. Removing any loss
 260 reduces performance, e.g., excluding fine-grained causal discrimination drops *Replace-Rel* by 1.5
 261 points (average 57.9). These results demonstrate that combining alignment, counterfactual, and causal
 262 objectives is crucial for compositional reasoning in multi-modal models.
 263

264 4.3 Effect of Counterfactual Supervision on Compositional Generalization

265 **Impact of Counterfactual Types on Task-specific Accuracy.** To further investigate the source
 266 of performance gains, we analyze the contribution of different types and quantities of counterfactual
 267 samples. The left of Figure 4 reports the results of selectively removing specific types of

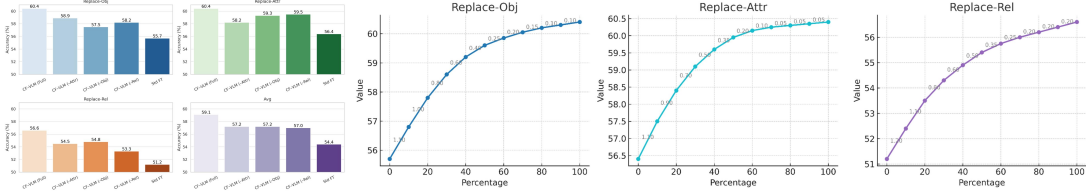


Figure 4: Removing any counterfactual type hurts task accuracy (left); more counterfactuals improve performance with diminishing returns (right).

counterfactual examples—namely, attribute-level (Attr), object-level (Obj), and relation-level (Rel) modifications—from CF-VLM trained with ViT-B/32 on the ConMe benchmark.

The results indicate that omitting any single category of counterfactuals leads to a noticeable performance drop on the corresponding subtask. For instance, removing attribute-level counterfactuals reduces accuracy on the *Replace-Attr* task by 2.2 points (from 60.4 to 58.2), while removing object- and relation-level samples results in decreases of 2.9 and 3.3 points on *Replace-Obj* and *Replace-Rel*, respectively. The overall average accuracy also declines accordingly.

Effect of Counterfactual Quantity: Tradeoff Between Performance and Efficiency. We further examine the effect of counterfactual data quantity on model performance. As shown in the right of Figure 4, the average accuracy on the ConMe benchmark increases steadily as the proportion of counterfactual samples (relative to the number of original positive samples) grows—from 0% (Standard Fine-tuning, 54.43%) to 100% (CF-VLM, 59.13%).

The performance curve begins to plateau between approximately 60% and 80% counterfactual ratio, suggesting diminishing marginal gains as more counterfactual data is introduced. This observation highlights a practical tradeoff: in real-world scenarios, one can balance performance improvement and computational cost by selecting an appropriate volume of counterfactual supervision.

4.4 Object-Level Hallucination Evaluation

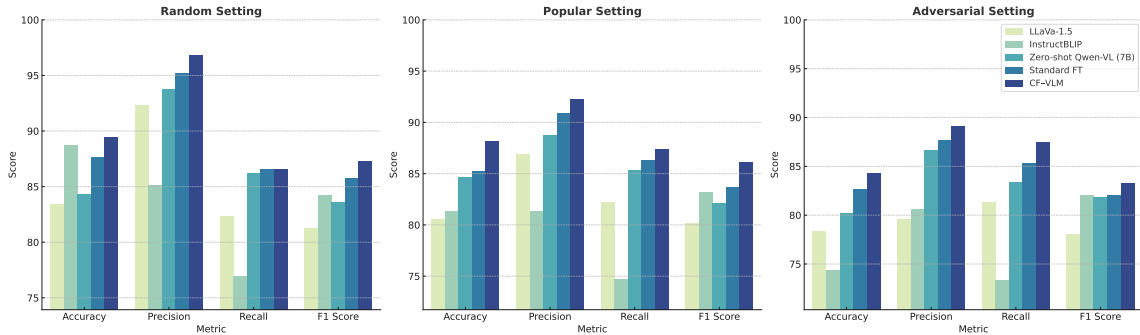


Figure 5: Hallucination evaluation on the POPE benchmark. CF-VLM improves accuracy, precision, recall, and F1 on the object existence task over baselines.

To further assess the practical applicability of CF-VLM, we evaluate its potential in mitigating visual hallucinations—a long-standing challenge for vision-language models. Although CF-VLM is not explicitly designed to optimize hallucination suppression, we hypothesize that its enhanced capability for fine-grained semantic modeling may indirectly improve factual alignment. To validate this hypothesis, we conduct evaluations on two widely adopted hallucination benchmarks: POPE [50] and MME [51].

As shown in Figure 5, on POPE’s object existence task, *Standard FT* (Qwen-VL, CC12M) significantly reduces the false positive rate compared to the zero-shot baseline, demonstrating the factual benefits of supervised fine-tuning. Building upon this, *CF-VLM* achieves marginal improvements across all four metrics—accuracy, precision, recall, and F1 score—with average gains of 0.5–1.2 percentage points.

MME benchmark results (Figure 6) show CF-VLM outperforming Standard FT on hallucination-sensitive subtasks (*Existence*, *Count*, *Position*, *Color*). Notable gains in *Color* (+2.0) and *Position*

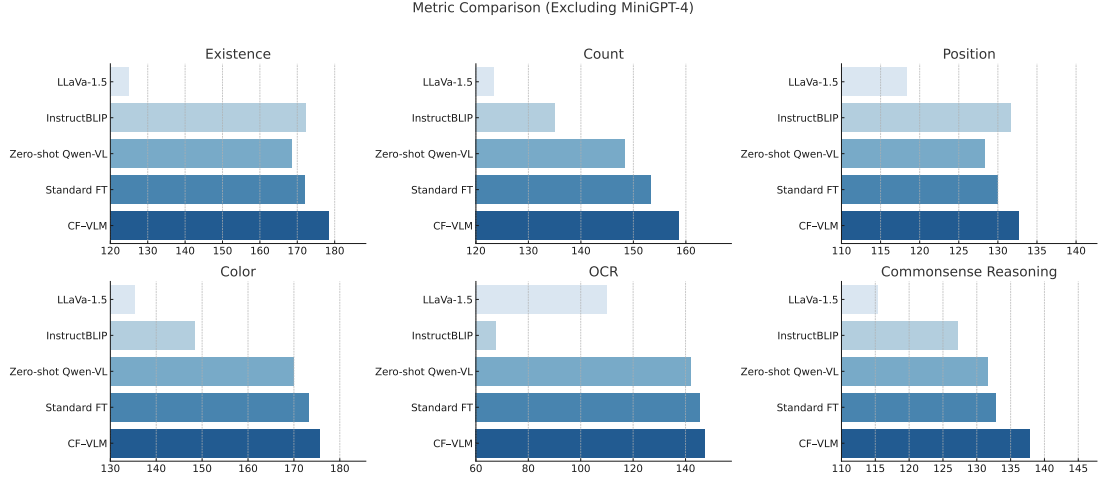


Figure 6: Hallucination evaluation on the MME benchmark. Relative performance gains on MME show CF-VLM’s advantage in hallucination-sensitive tasks, especially for color and position, indicating enhanced semantic grounding via counterfactual training.

(+1.4) highlight counterfactual supervision’s benefit for attribute-level reasoning, likely due to minimal perturbations in attribute representations reducing hallucination.

5 Related Work

Recent advances in vision-language models (VLMs)[52–54], such as CLIP [6] and BLIP [1], have demonstrated remarkable performance in cross-modal tasks through large-scale image-text pretraining. However, these models often struggle with fine-grained discrimination and causal reasoning, relying on superficial statistical correlations rather than capturing deep causal relationships between visual and textual content [5, 7]. Existing approaches employ contrastive learning to enhance embedding space discriminability, such as triplet loss[17], but primarily focus on binary matching, overlooking the impact of causal attribute changes on semantics [19].

Counterfactual learning has emerged as a promising approach to address the causal reasoning limitations of VLMs[18, 55, 56]. introduced counterfactual visual explanations by modifying key visual elements to analyze model decisions, while CPL [19] leveraged counterfactual prompts to improve cross-modal understanding. However, these methods typically focus on single-modality counterfactuals, lacking joint image-text counterfactual designs [57] and using singular objectives that fail to leverage counterfactual samples for multi-level semantic supervision.

Our CF-VLM framework advances VLM causal reasoning by integrating joint image-text counterfactual samples with three complementary training objectives: cross-modal alignment, counterfactual scenario discrimination, and fine-grained causal discrimination. Unlike TripletCLIP [16], which optimizes coarse-grained similarity, CF-VLM targets causal decision points, achieving superior accuracy on benchmarks like ARO, ConMe, and VL-Checklist, while reducing visual hallucinations, demonstrating enhanced robustness and applicability in high-stakes scenarios.

6 Conclusion and Limitations

We propose CF-VLM, a counterfactual fine-tuning framework to enhance semantic granularity and causal reasoning in vision-language models (VLMs). Addressing limitations in distinguishing subtle semantic differences, CF-VLM uses minimally edited image-text pairs from a fine-tuned SDXL pipeline, introducing counterfactual scenario discrimination and fine-grained causal discrimination objectives. Evaluations on compositional reasoning (ARO, ConMe, VL-Checklist) and hallucination benchmarks (POPE, MME) show CF-VLM outperforming CLIP-based and LLM-augmented VLMs, improving factual grounding and reducing hallucinations, even on untrained benchmarks, demonstrating the generality of counterfactual supervision. Future work includes integrating human-in-the-loop editing for richer counterfactuals, applying CF-VLM to tasks like VQA or image editing, and exploring interpretability via counterfactual sensitivity for more transparent VLMs.

References

- [1] J. Li, D. Li, C. Xiong, and S. Hoi, “Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation,” 2022. [Online]. Available: <https://arxiv.org/abs/2201.12086>
- [2] Y. Du, Z. Liu, J. Li, and W. X. Zhao, “A survey of vision-language pre-trained models,” 2022. [Online]. Available: <https://arxiv.org/abs/2202.10936>
- [3] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman, and K. Simonyan, “Flamingo: a visual language model for few-shot learning,” 2022.
- [4] J. Ma, “Causal inference with large language model: A survey,” 2025. [Online]. Available: <https://arxiv.org/abs/2409.09822>
- [5] M. Chen, B. Peng, Y. Zhang, and C. Lu, “Cello: Causal evaluation of large vision-language models,” 2024. [Online]. Available: <https://arxiv.org/abs/2406.19131>
- [6] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, “Learning transferable visual models from natural language supervision,” 2021. [Online]. Available: <https://arxiv.org/abs/2103.00020>
- [7] E. Bugliarello, L. Sartran, A. Agrawal, L. A. Hendricks, and A. Nematzadeh, “Measuring progress in fine-grained vision-and-language understanding,” in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 1559–1582. [Online]. Available: <https://aclanthology.org/2023.acl-long.87/>
- [8] W. Peng, S. Xie, Z. You, S. Lan, and Z. Wu, “Synthesize, diagnose, and optimize: Towards fine-grained vision-language understanding,” in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 13 279–13 288.
- [9] J. Kim and H. Ji, “Finer: Investigating and enhancing fine-grained visual concept recognition in large vision language models,” in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds. Miami, Florida, USA: Association for Computational Linguistics, Nov. 2024, pp. 6187–6207. [Online]. Available: <https://aclanthology.org/2024.emnlp-main.356/>
- [10] Z. Xu, Y. Zhu, S. Deng, A. Mittal, Y. Chen, M. Wang, P. Favaro, J. Tighe, and D. Modolo, “Benchmarking zero-shot recognition with vision-language models: Challenges on granularity and specificity,” in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2024, pp. 1827–1836.
- [11] W. Wang, Z. Yang, B. Xu, J. Li, and Y. Sun, “Vilta: Enhancing vision-language pre-training through textual augmentation,” in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 3135–3146.
- [12] V. Palit, R. Pandey, A. Arora, and P. P. Liang, “Towards vision-language mechanistic interpretability: A causal tracing tool for blip,” in *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2023, pp. 2848–2853.
- [13] H. Laurençon, L. Tronchon, M. Cord, and V. Sanh, “What matters when building vision-language models?” 2024. [Online]. Available: <https://arxiv.org/abs/2405.02246>
- [14] M. Wahed, X. Zhou, T. Yu, and I. Lourentzou, “Fine-grained alignment for cross-modal recipe retrieval,” in *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2024, pp. 5572–5581.
- [15] A. D’Innocente, N. Garg, Y. Zhang, L. Bazzani, and M. Donoser, “Localized triplet loss for fine-grained fashion image retrieval,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021, pp. 3905–3910.
- [16] M. Patel, A. Kusumba, S. Cheng, C. Kim, T. Gokhale, C. Baral, and Y. Yang, “Tripletclip: Improving compositional reasoning of clip via synthetic vision-language negatives,” 2024. [Online]. Available: <https://arxiv.org/abs/2411.02545>
- [17] F. Schroff, D. Kalenichenko, and J. Philbin, “Facenet: A unified embedding for face recognition and clustering,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Jun. 2015, p. 815–823. [Online]. Available: <http://dx.doi.org/10.1109/CVPR.2015.7298682>

- [18] Y. Rao, G. Chen, J. Lu, and J. Zhou, “Counterfactual attention learning for fine-grained visual categorization and re-identification,” in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 1005–1014.
- [19] X. He, D. Yang, W. Feng, T.-J. Fu, A. Akula, V. Jampani, P. Narayana, S. Basu, W. Y. Wang, and X. Wang, “CPL: Counterfactual prompt learning for vision and language models,” in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec. 2022, pp. 3407–3418. [Online]. Available: <https://aclanthology.org/2022.emnlp-main.224/>
- [20] Y. Goyal, Z. Wu, J. Ernst, D. Batra, D. Parikh, and S. Lee, “Counterfactual visual explanations,” 2019. [Online]. Available: <https://arxiv.org/abs/1904.07451>
- [21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” 2023. [Online]. Available: <https://arxiv.org/abs/1706.03762>
- [22] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, W. Ye, Y. Zhang, Y. Chang, P. S. Yu, Q. Yang, and X. Xie, “A survey on evaluation of large language models,” *ACM Trans. Intell. Syst. Technol.*, vol. 15, no. 3, Mar. 2024. [Online]. Available: <https://doi.org/10.1145/3641289>
- [23] Y. Liu, Y. Zhang, Y. Wang, F. Hou, J. Yuan, J. Tian, Y. Zhang, Z. Shi, J. Fan, and Z. He, “A survey of visual transformers,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 6, pp. 7478–7498, 2024.
- [24] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A simple framework for contrastive learning of visual representations,” in *Proceedings of the 37th International Conference on Machine Learning*, ser. ICML’20. JMLR.org, 2020.
- [25] T. Wang and P. Isola, “Understanding contrastive representation learning through alignment and uniformity on the hypersphere,” in *Proceedings of the 37th International Conference on Machine Learning*, ser. ICML’20. JMLR.org, 2020.
- [26] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. V. Le, Y. Sung, Z. Li, and T. Duerig, “Scaling up visual and vision-language representation learning with noisy text supervision,” 2021. [Online]. Available: <https://arxiv.org/abs/2102.05918>
- [27] N. Kim and T. Linzen, “Cogs: A compositional generalization challenge based on semantic interpretation,” 2020. [Online]. Available: <https://arxiv.org/abs/2010.05465>
- [28] D. Teney, E. Abbasnedjad, and A. van den Hengel, “Learning what makes a difference from counterfactual examples and gradient supervision,” in *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X*. Berlin, Heidelberg: Springer-Verlag, 2020, p. 580–599. [Online]. Available: https://doi.org/10.1007/978-3-030-58607-2_34
- [29] R. Paiss, A. Ephrat, O. Tov, S. Zada, I. Mosseri, M. Irani, and T. Dekel, “Teaching clip to count to ten,” in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 3147–3157.
- [30] Z. Chen, Q. Gao, A. Bosselut, A. Sabharwal, and K. Richardson, “Disco: Distilling counterfactuals with large language models,” 2023. [Online]. Available: <https://arxiv.org/abs/2212.10534>
- [31] Y. Zhang, J. Wang, and J. Sang, “Counterfactually measuring and eliminating social bias in vision-language pre-training models,” in *Proceedings of the 30th ACM International Conference on Multimedia*, ser. MM ’22. New York, NY, USA: Association for Computing Machinery, 2022, p. 4996–5004. [Online]. Available: <https://doi.org/10.1145/3503161.3548396>
- [32] Y. Rao, G. Chen, J. Lu, and J. Zhou, “Counterfactual attention learning for fine-grained visual categorization and re-identification,” in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 1005–1014.
- [33] L. Chen, Y. Zheng, Y. Niu, H. Zhang, and J. Xiao, “Counterfactual samples synthesizing and training for robust visual question answering,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 11, pp. 13 218–13 234, 2023.
- [34] Qwen, :, A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, H. Lin, J. Yang, J. Tu, J. Zhang, J. Yang, J. Yang, J. Zhou, J. Lin, K. Dang, K. Lu, K. Bao, K. Yang, L. Yu, M. Li, M. Xue, P. Zhang, Q. Zhu, R. Men, R. Lin, T. Li, T. Tang, T. Xia, X. Ren, X. Ren, Y. Fan, Y. Su, Y. Zhang, Y. Wan, Y. Liu, Z. Cui, Z. Zhang, and Z. Qiu, “Qwen2.5 technical report,” 2025. [Online]. Available: <https://arxiv.org/abs/2412.15115>

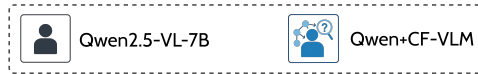
- [35] S. Changpinyo, P. Sharma, N. Ding, and R. Soricut, “Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts,” in *CVPR*, 2021.
- [36] P. Sharma, N. Ding, S. Goodman, and R. Soricut, “Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning,” in *Proceedings of ACL*, 2018.
- [37] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollar, and C. L. Zitnick, “Microsoft coco captions: Data collection and evaluation server,” 2015. [Online]. Available: <https://arxiv.org/abs/1504.00325>
- [38] I. Huang, W. Lin, M. J. Mirza, J. A. Hansen, S. Doveh, V. I. Butoi, R. Herzig, A. Arbel, H. Kuehne, T. Darrell, C. Gan, A. Oliva, R. Feris, and L. Karlinsky, “Conme: Rethinking evaluation of compositional reasoning for modern vlms,” 2024. [Online]. Available: <https://arxiv.org/abs/2406.08164>
- [39] M. Yuksekgonul, F. Bianchi, P. Kalluri, D. Jurafsky, and J. Zou, “When and why vision-language models behave like bags-of-words, and what to do about it?” in *International Conference on Learning Representations*, 2023. [Online]. Available: <https://openreview.net/forum?id=KRLUvvh8uaX>
- [40] T. Zhao, T. Zhang, M. Zhu, H. Shen, K. Lee, X. Lu, and J. Yin, “VL-checklist: Evaluating pre-trained vision-language models with objects, attributes and relations,” 2023. [Online]. Available: <https://arxiv.org/abs/2207.00221>
- [41] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, “ImageNet Large Scale Visual Recognition Challenge,” *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.
- [42] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, “From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions,” *Transactions of the Association for Computational Linguistics*, vol. 2, pp. 67–78, 2014.
- [43] L. Zhang, R. Awal, and A. Agrawal, “Contrasting intra-modal and ranking cross-modal hard negatives to enhance visio-linguistic compositional understanding,” 2024. [Online]. Available: <https://arxiv.org/abs/2306.08832>
- [44] F. Parascandolo, N. Moratelli, E. Sangineto, L. Baraldi, and R. Cucchiara, “Causal graphical models for vision-language compositional understanding,” 2025. [Online]. Available: <https://arxiv.org/abs/2412.09353>
- [45] Y. Huang, J. Tang, Z. Chen, R. Zhang, X. Zhang, W. Chen, Z. Zhao, Z. Zhao, T. Lv, Z. Hu, and W. Zhang, “Structure-clip: Towards scene graph knowledge to enhance multi-modal structured representations,” 2023. [Online]. Available: <https://arxiv.org/abs/2305.06152>
- [46] H. Liu, C. Li, Y. Li, and Y. J. Lee, “Improved baselines with visual instruction tuning,” 2024. [Online]. Available: <https://arxiv.org/abs/2310.03744>
- [47] W. Dai, J. Li, D. Li, A. M. H. Tiong, J. Zhao, W. Wang, B. Li, P. Fung, and S. Hoi, “Instructblip: Towards general-purpose vision-language models with instruction tuning,” 2023. [Online]. Available: <https://arxiv.org/abs/2305.06500>
- [48] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, “Minigpt-4: Enhancing vision-language understanding with advanced large language models,” 2023. [Online]. Available: <https://arxiv.org/abs/2304.10592>
- [49] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, “Sdxl: Improving latent diffusion models for high-resolution image synthesis,” 2023. [Online]. Available: <https://arxiv.org/abs/2307.01952>
- [50] Y. Li, Y. Du, K. Zhou, J. Wang, W. X. Zhao, and J.-R. Wen, “Evaluating object hallucination in large vision-language models,” 2023. [Online]. Available: <https://arxiv.org/abs/2305.10355>
- [51] C. Fu, P. Chen, Y. Shen, Y. Qin, M. Zhang, X. Lin, J. Yang, X. Zheng, K. Li, X. Sun, Y. Wu, and R. Ji, “Mme: A comprehensive evaluation benchmark for multimodal large language models,” 2024. [Online]. Available: <https://arxiv.org/abs/2306.13394>
- [52] S. Long, F. Cao, S. C. Han, and H. Yang, “Vision-and-language pretrained models: A survey,” 2022. [Online]. Available: <https://arxiv.org/abs/2204.07356>
- [53] A. Ghosh, A. Acharya, S. Saha, V. Jain, and A. Chadha, “Exploring the frontier of vision-language models: A survey of current methodologies and future directions,” 2024. [Online]. Available: <https://arxiv.org/abs/2404.07214>

- 484 [54] J. Zhang, J. Huang, S. Jin, and S. Lu, "Vision-language models for vision tasks: A survey," *IEEE*
485 *Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 8, pp. 5625–5644, 2024.
- 486 [55] E. Abbasnejad, D. Teney, A. Parvaneh, J. Shi, and A. van den Hengel, "Counterfactual vision and language
487 learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp.
488 10 041–10 051.
- 489 [56] M.-K. Xie, J.-H. Xiao, P. Peng, G. Niu, M. Sugiyama, and S.-J. Huang, "Counterfactual reasoning for
490 multi-label image classification via patching-based training," in *Proceedings of the 41st International*
491 *Conference on Machine Learning*, ser. ICML'24. JMLR.org, 2024.
- 492 [57] F. Parascandolo, N. Moratelli, E. Sangineto, L. Baraldi, and R. Cucchiara, "Causal graphical models for
493 vision-language compositional understanding," 2025. [Online]. Available: <https://arxiv.org/abs/2412.09353>
- 494 [58] X. He, D. Yang, W. Feng, T.-J. Fu, A. Akula, V. Jampani, P. Narayana, S. Basu, W. Y. Wang, and X. E.
495 Wang, "Cpl: Counterfactual prompt learning for vision and language models," 2022. [Online]. Available:
496 <https://arxiv.org/abs/2210.10362>
- 497 [59] Y. Goyal, Z. Wu, J. Ernst, D. Batra, D. Parikh, and S. Lee, "Counterfactual visual explanations," 2019.
498 [Online]. Available: <https://arxiv.org/abs/1904.07451>
- 499 [60] Y. Rao, G. Chen, J. Lu, and J. Zhou, "Counterfactual attention learning for fine-grained visual
500 categorization and re-identification," 2021. [Online]. Available: <https://arxiv.org/abs/2108.08728>

A Dataset Details

Our training and evaluation are conducted on a series of standardized image-text datasets and structured evaluation benchmarks, as detailed below:

- **Training Data:** We utilize the cleaned version of CC12M (8.6M image-text pairs) and its subset CC3M (2.6M pairs) as our primary training corpora. All samples undergo standardized preprocessing, including image resizing to 224×224 , text normalization, and language alignment to ensure stable input distribution.
- **Counterfactual Sample Generation:** For each original image-text pair, we generate a semantically similar but minimally perturbed counterfactual pair using our Dynamic Counterfactual Generation (DCF) strategy. This effectively doubles the training data and enhances the model’s capacity to learn fine-grained semantic variations through contrastive supervision.
- **MSCOCO Captions (120K pairs):** This dataset is not used for training, but is optionally included for auxiliary analysis tasks (e.g., image-text retrieval or structural generalization), to assess the transferability of our method beyond the core training corpus.
- **Evaluation Benchmarks:** We evaluate the model’s compositional reasoning and cross-modal generalization on the following benchmarks:
 - **ConMe Suite:** This benchmark focuses on the model’s ability to distinguish subtle semantic changes in compositional logic, such as negation, sequence, and conjunction. Each sample presents an image with two candidate captions—one semantically correct and one minimally perturbed. The task evaluates causal reasoning and structural understanding.
 - **ARO (Attribute–Relation–Object):** This benchmark introduces targeted perturbations to image-text pairs along three dimensions: attributes, relationships, and object identity. It assesses the model’s consistency in recognizing multi-dimensional semantic alignment, particularly in fine-grained visual-linguistic scenarios.
 - **VL-Checklist:** This benchmark contains three sub-tasks—Replace-Obj, Replace-Attr, and Replace-Rel—where only one semantic element in the text is systematically modified. It is designed to evaluate the model’s sensitivity to attribute- and relation-level counterfactuals.
 - **ImageNet-1k (Zero-shot Classification):** This task transforms the standard ImageNet classification problem into an image-text matching setup. Each image is paired with multiple candidate textual descriptions, and the model must select the most semantically aligned one. This evaluates the model’s ability to generalize to natural image distributions in a zero-shot setting.
 - **MSCOCO / Flickr30k (Image-Text Retrieval):** These retrieval tasks involve bi-directional matching (image \rightarrow text and text \rightarrow image). We report Recall@1 and Recall@5 metrics to evaluate whether the model can accurately locate the correct match from a large candidate pool.
- **Preprocessing and Splits:** All datasets follow their official train/validation/test splits. During training, each batch maintains a fixed 1:4 ratio between factual and counterfactual samples, to enhance the model’s sensitivity to semantically critical perturbations.



A woman in a wheelchair is sitting next to a **parking garage** . ❌

A woman in a wheelchair is sitting next to a **wall with the word 'park' on it** . ✅



A man is **playing with his dog** in the park . ❌

A man is playing with his dog in the park , and the **dog is jumping to catch a frisbee** . ✅



A **brown dog** is running on the porch of a house . ❌

A **golden retriever** is running on the porch of a house . ✅



Two people are **hiking in the woods** near a waterfall . ❌

Two people are **standing on a mossy path** near a waterfall , ✅



A young boy is riding a bicycle while a young girl in a **light blue dress** watches . ❌

A young girl in a **green dress** stands next to a man and a boy on a bicycle . ✅



Two women sit on a train and **look out the window** . ❌

Two women sit on a train , one **reading** and the other looking out the window . ✅



A dog is **standing** in the grass and looking up . ❌

A dog is **running** on the grass with its mouth open . ✅



A woman is **pushing her daughter** on a swing in a park . ❌

A man is helping a young girl to **cross a rope bridge** in a park . ✅



A man in a white shirt and blue jeans is riding a **blue tricycle** . ❌

A man in a white shirt and blue jeans is **riding a blue bike** . ✅



A group of people are **playing a video game** in a room . ❌

A group of people are **dancing** in the middle of a room with other people sitting around them . ✅



A woman in a white shirt and blue shorts is **playing tennis with a man** in a blue shirt and red shorts . ❌

A woman in a white shirt and blue shorts is playing tennis **while a man in a blue shirt and red shorts watches** . ✅



A young **girl** in a red shirt is running with **her dog** on the grass . ❌

A young **boy** in a red shirt and **gray pants** is running with **his dog** . ✅

Figure 7: Qualitative comparisons showing CF-VLM's improvements in fine-grained attribute recognition, causal action modeling, and relational understanding across diverse visual scenes.

Counterfactual Rewriting Prompt

You are a rewriting engine specialized in generating counterfactual or counter-attribute descriptions.

Your task: Rewrite the input sentence by changing **exactly one element** — either:

- a **physical attribute of an object**, or
- a **causal relationship**.

You may change one of the following object attributes (be creative and visually expressive):

- Color (e.g., red → blue)
- Material (e.g., wood → metal)
- Shape (e.g., round → square)
- Size (e.g., large → small)
- Position or orientation (e.g., on the left → on the right)
- State or pose (e.g., standing → lying)
- Object category (e.g., dog → cat)
- Parts or structure (e.g., three legs → four legs)
- Movement status (e.g., jumping → still)
- Quantity (e.g., two → one)
- Facial expression or emotion (e.g., smiling → angry)

You may use imaginative or creative combinations as long as they are visually descriptive and logically coherent.

Alternatively, you may flip a causal relationship:

1. Identify a cause-effect link in the sentence.
2. Invert it: If cause A didn't happen, effect B would change.
3. Rewrite the sentence naturally under the new causal logic.

Rules:

- Modify **only one thing** (either one attribute or one causal link).
- Keep all other elements in the sentence **exactly the same**.
- Do **not** repeat any previously generated sentence.
- The output must be a **single fluent, grammatically correct sentence**.
- **Do not** include any explanation, comment, tag, or prefix.
- **Only** return the rewritten sentence. Nothing else.

Format: Please output the K counterfactual sentences in a numbered list:

1. [sentence #1]
2. [sentence #2]
3. [sentence #3]

...

Example Input:

A young woman holding a racket hit the ball, and the ball flew outward.

Output (attribute):

1. An old woman holding a racket hit the ball, and the ball flew outward.
2. A transparent woman holding a racket hit the ball, and the ball flew outward.

Output (causal):

3. A young woman holding a racket missed the ball, and the ball dropped to the ground.

Now return K distinct counterfactual rewrites in the required format.

B Qualitative Analysis of Post-training Performance

B.1 Diversity and Quality Control of Counterfactual Samples

CF-VLM employs a structured counterfactual editing strategy that includes four types of controlled interventions: attribute substitution, object substitution, relationship substitution, and location substitution. As illustrated in Figure 8, attribute edits (e.g., replacing “dirt” with “paved roads”) and object edits (e.g., changing “two men” to “two women”) enable the model to learn from variations in properties and subject identities. Meanwhile, relationship edits (e.g., switching from “throwing a pitch” to “catching a ball”) and location edits (e.g., changing the spatial configuration between a “cat” and a “television”) enhance the model’s understanding of action sequences and spatial semantics.

These controlled counterfactuals enable CF-VLM to cover a wider range of semantic types—from color and texture to interaction and spatial reasoning—while avoiding issues commonly observed in large language model (LLM)-based generation, such as redundancy and logical inconsistency. This structured supervision improves both the semantic diversity and logical coherence of the training samples.

Figure 7 provides qualitative evidence supporting this claim. In the top-left example, the baseline model incorrectly identifies the scene as “a parking garage” and ignores the textual cue “park” on the wall, while CF-VLM accurately describes the scene with attribute-aware recognition. In the second-left row, CF-VLM enhances the original caption “a man is playing with his dog” by adding “and the dog is jumping to catch a frisbee,” capturing the entire action-outcome chain. Similarly, in the top-right image, the baseline produces a generic “brown dog,” while CF-VLM correctly identifies it as a “golden retriever,” offering greater specificity. In another example, CF-VLM replaces “hiking near a waterfall” with “standing on a mossy path near a waterfall,” yielding more grounded spatial reasoning. These examples demonstrate CF-VLM’s clear advantage in fine-grained attribute recognition, action completion, and relationship understanding. The generated counterfactuals are both more diverse and semantically consistent than those produced by conventional methods.

B.2 Causal Depth

CF-VLM also demonstrates superior causal reasoning by explicitly modeling the structural components of visual events, including agents, objects, action stages, and outcome states. During training, counterfactual interventions are applied in a structured manner, enabling the model to learn coherent “intervention-effect” causal chains.

As shown in Figure 7, CF-VLM captures rich event semantics across a range of complex scenes. In the “man-dog-frisbee” scenario, CF-VLM identifies a three-step causal sequence: “a man is playing with his dog” → “the dog is jumping to catch a frisbee,” exhibiting strong temporal and causal coherence. In contrast, the baseline merely describes surface-level actions without modeling the outcome.

In the “tennis” example, CF-VLM accurately distinguishes between the active player and the observer, generating the caption “a woman is playing tennis while a man watches,” thus capturing role-based causal dynamics. In other scenes—such as helping a child cross a bridge or shifting spatial relationships—CF-VLM successfully tracks pre- and post-action semantic states, encoding progression and continuity within the event timeline.

These results indicate that CF-VLM not only improves action recognition but also accurately models temporal structures and agent-object interactions. Overall, CF-VLM outperforms conventional baselines in understanding complex visual events, capturing semantically consistent changes triggered by causal interventions. This structured causal modeling provides a stronger foundation for generalization and discrimination in multimodal semantic tasks, validating the effectiveness of counterfactual supervision for compositional reasoning.

C Detailed Comparison with Related Work and Highlighting Unique Contributions

In order to clarify CF-VLM’s unique contributions in enhancing fine-grained discrimination and advancing deeper causal understanding in vision-language models (VLMs), and to directly address

concerns about novelty, this section presents a thorough comparison of CF-VLM against several key related works at the core mechanism level. We aim to demonstrate that CF-VLM is not a simple application of counterfactual samples but a carefully designed, multifaceted framework to improve sensitivity to specific interventions and identify causal visual elements.

C.1 Mechanistic-Level Comparison

We compare CF-VLM and representative works—including TripletCLIP, CPL [58], Goyal et al. [59], and Rao et al. [60]—along four dimensions: the type and generation of counterfactual samples, core learning objective/loss design, focus on causality or fine-grained discrimination, and simultaneous handling of image and text counterfactuals.

Type and Generation of Counterfactual Samples CF-VLM adopts a comprehensive strategy for constructing counterfactuals: it combines *complete counterfactual scenarios* where both image and text are jointly edited, with *minimally edited* image perturbations targeting single object attributes or causal relations (e.g., color, class, state). We use fine-tuned SDXL 1.0 for image generation and Qwen2-72B-Instruct for text editing to ensure semantic control.

In contrast:

- **TripletCLIP[16]** synthesizes visual–language negatives via rule- or template-based text perturbations to improve compositional reasoning.
- **CPL [58]** generates counterfactuals by modifying textual prompts using predefined templates to enhance cross-modal robustness.
- **Goyal et al. [59]** focuses on image-based edits of key elements for interpretability, not directly optimizing prediction performance.
- **Rao et al. [60]** learns counterfactual attention by masking or replacing salient regions, calibrating attention maps for visual tasks.

Core Learning Objectives and Loss Design CF-VLM’s training objective integrates three complementary losses:

1. **Cross-modal Alignment Loss** L_{align} maintains baseline image–text matching and prevents forgetting.
2. **Counterfactual Scene Distinction Loss** L_{csd} contrasts factual versus complete counterfactual scenarios to reinforce representational uniqueness.
3. **Fine-grained Causal Distinction Loss** L_{fcd} sharpens sensitivity to minimal but critical causal edits.

Other works typically use a single or simpler combination of objectives: TripletCLIP employs a standard triplet loss; CPL uses contrastive learning on textual prompts; Goyal et al. and Rao et al. use counterfactuals primarily for interpretability or attention robustness rather than end-to-end optimization.

Focus on Causality and Fine-Grained Discrimination CF-VLM explicitly targets sensitivity to controlled local interventions (“causal decision points”) that determine image–text matching. Unlike TripletCLIP’s focus on compositional negative mining or CPL’s OOD robustness objective, CF-VLM trains the model to understand why subtle changes alter semantics. Goyal et al. emphasizes interpretability and Rao et al. enhances attention robustness but neither systemically models causal edits tied to fine-grained semantic shifts. **Simultaneous Handling of Image and Text Counterfactuals** CF-VLM uniquely processes both jointly edited image–text scenarios in L_{csd} and pairs minimally edited images with original text in L_{fcd} . TripletCLIP and CPL focus on text-only counterfactuals, while Goyal et al. and Rao et al. operate primarily on image perturbations. CF-VLM’s dual-modality approach provides richer, multi-level supervisory signals.

C.2 Analysis of CF-VLM’s Uniqueness and Integrative Advantages

CF-VLM’s distinctive integrative strengths can be summarized as:

1. **Diverse, Controlled Counterfactuals:** Combines high-quality minimal-edits and complete scenarios to offer multi-level contrast.
2. **Multi-Objective Synergy:** L_{align} , L_{csd} , and L_{fed} work together to maintain alignment, distinguish scenes, and detect causal edits.
3. **Deep Focus on Causal Decision Points:** Guides the model to learn why small edits affect semantics, advancing causal understanding beyond feature matching.
4. **End-to-End, Controllable Fine-Tuning Framework:** Integrates SDXL and LLM-based generation into a deployable pipeline to retrofit existing VLMs.

D Detailed Experimental Setup and Hyperparameters

This section provides a comprehensive overview of the experimental setup, including hardware and software configurations, dataset preparation specifics, and detailed hyperparameter settings used for training and evaluating our proposed **CF-VLM** framework and all baseline models.

D.1 General Setup

All experiments were conducted on a consistent hardware and software environment to ensure fair comparisons.

Hardware: Experiments were primarily run on NVIDIA A100 GPUs. Depending on model size and batch configuration, between 1 to 8 GPUs were utilized per experimental run. **Software:** Key software libraries included Python 3.9+, PyTorch 1.13.1 (with CUDA 11.7), Transformers 4.28.1, and a customized fork of OpenCLIP for certain baseline implementations. Standard scientific computing libraries such as NumPy 1.23.5 and Pandas 1.5.3 were used for data handling and analysis. **Operating System:** All nodes ran Ubuntu 20.04 LTS.

D.2 Dataset Preprocessing and Splits

Datasets including CC12M, CC3M, MSCOCO Captions, ARO, Conne, VL-Checklist, ImageNet-1k, and Flickr30k were used. For CC12M and CC3M, we followed standard filtering protocols similar to those described in prior large-scale VLM pretraining works, removing images with insufficient resolution, near-duplicate captions, and potential NSFW content based on automated filters. For pretraining/fine-tuning on CC12M/CC3M, we used a 99% training and 1% validation split, randomly sampled. For evaluation on downstream tasks, we adhered to their standard publicly available splits. For ImageNet-1k, standard validation set images were used for zero-shot classification. For MSCOCO and Flickr30k, we utilized the Karpathy splits for zero-shot retrieval.

D.3 Hyperparameter Settings for CF-VLM Fine-tuning

Our CF-VLM framework was fine-tuned on pretrained Qwen2.5-VL (7B) and LLaVA-1.5 (7B/13B variants where applicable) models. The core hyperparameters for CF-VLM fine-tuning are detailed in Table 5.

The selection of loss weights (α, β, γ) and margins (m_1, m_2) was performed via grid search on a small subset of the CC3M validation data, optimizing for average performance Improvement on key compositional reasoning benchmarks (ARO, Conne-Avg). The temperature τ for L_{align} was generally kept consistent with standard CLIP pretraining practices or the original VLM’s configuration.

D.4 Baseline Model Hyperparameters

For all baseline models (e.g., Zero-shot, Standard Fine-tuning, Text-Negative Fine-tuning, Triplet-CLIP*, CE-CLIP+, etc.), we endeavored to reproduce their results using publicly available codebases and reported hyperparameters where available. For our own Standard Fine-tuning (Std FT) and Text-Negative Fine-tuning (TextNeg FT) baselines applied to Qwen-VL and LLaVA-1.5, we used a setup similar to CF-VLM’s L_{align} component, with identical optimizer settings, learning rates, and batch sizes to ensure fair comparison. Key differences are outlined below:

Table 5: Core hyperparameter settings for CF-VLM fine-tuning. Values were kept consistent across different base models (Qwen-VL, LLaVA-1.5) unless specified. Slight variations (e.g., in learning rate or batch size) were explored during initial tuning, with the reported values yielding optimal performance on a held-out validation set derived from CC3M.

Hyperparameter	Value / Range
Optimizer	AdamW
AdamW β_1	0.9
AdamW β_2	0.98 (Qwen-VL), 0.985 (LLaVA-1.5)
AdamW ϵ	1×10^{-6} (Qwen-VL), 1×10^{-7} (LLaVA-1.5)
Peak Learning Rate (lr_{peak})	1×10^{-5} (for 7B models), 8×10^{-6} (for 13B models)
Learning Rate Schedule	Cosine decay with linear warmup
Warmup Steps	500 (CC12M), 300 (CC3M)
Weight Decay	0.1 (Qwen-VL), 0.08 (LLaVA-1.5)
Effective Batch Size	256 (distributed across GPUs)
Micro-batch Size per GPU	Varied (4 to 16, depending on GPU memory and model)
Gradient Accumulation Steps	Adjusted to achieve effective batch size
Training Steps (CC12M+DCF)	200,000
Training Steps (CC3M+DCF)	90,000
Mixed Precision	BF16
Gradient Clipping Norm	1.0
Dropout (where applicable in VLM head)	0.1
Image Resolution	224x224 (CLIP-based), 336x336 or 448x448 (for Qwen-VL/LLaVA, model dependent)
CF-VLM Specific Loss Parameters	
L_{align} Temperature (τ)	0.07 (initial, learnable for some CLIP backbones, fixed for LLM-based VLMs)
L_{csd} Margin (m_1)	0.25
L_{fcd} Margin (m_2)	0.30
Loss Weight α (for L_{align})	1.0
Loss Weight β (for L_{csd})	0.45
Loss Weight γ (for L_{fcd})	0.55
Counterfactual Sample Ratio (in batch)	1:1 (Factual : Counterfactual)

- **Zero-shot Baselines:** Evaluated directly using the official pretrained checkpoints without any fine-tuning.
- **Standard Fine-tuning (Std FT):** Used only the L_{align} loss component with factual image-text pairs from CC12M or CC3M. Hyperparameters mirrored those in Table 5 for optimizer, LR, batch size, etc.
- **Text-Negative Fine-tuning (TextNeg FT):** Similar to Std FT, but for each factual pair (I, T) , a hard text negative T_{neg} (randomly sampled from other texts in the batch, or generated via simple rule-based negation for some experiments) was used to augment the contrastive loss. The negative sampling strategy and loss formulation followed common practices in the literature.
- **Other Reported Baselines (e.g., TripletCLIP*, COGT-CLIP):** We report scores directly from the respective publications or their official repositories. If re-running was necessary for a specific component (e.g., CLIP-ViT-B/32 on CC12M), we followed their reported settings as closely as possible. For instance, TripletCLIP* variants often use a margin of 0.2 for their triplet loss. NegCLIP typically involves in-batch negatives or a more sophisticated negative cache.

A summary of key training parameters for our implemented baselines is presented in Table 6.

D.5 Counterfactual Sample Generation Parameters

The generation of counterfactual text descriptions was performed by LLaMA-3-70B-Instruct using 3-shot chain-of-thought prompting. Counterfactual images were generated using SDXL 1.0 Base + Refiner with a 40+15 step schedule. For each factual image-text pair, four counterfactual samples (either image or text modified, or both for complete counterfactual scenarios) were generated, maintaining a 1:4 ratio of factual to counterfactual data during CF-VLM training stages that utilize them.

Table 6: Key hyperparameters for our implemented baseline fine-tuning experiments (Std FT, TextNeg FT). These largely mirror the CF-VLM settings for core optimizer and learning rate parameters to ensure comparability.

Hyperparameter	Std FT (Qwen-VL/LLaVA)	TextNeg FT (Qwen-VL/LLaVA)
Base Model	Qwen-VL (7B), LLaVA-1.5 (7B/13B)	Qwen-VL (7B), LLaVA-1.5 (7B/13B)
Optimizer	AdamW	AdamW
AdamW $\beta_1, \beta_2, \epsilon$	Same as CF-VLM (Table 5)	Same as CF-VLM (Table 5)
Peak Learning Rate	Same as CF-VLM	Same as CF-VLM
LR Schedule	Cosine decay with warmup	Cosine decay with warmup
Warmup Steps	Same as CF-VLM	Same as CF-VLM
Weight Decay	Same as CF-VLM	Same as CF-VLM
Effective Batch Size	256	256
Training Data	CC12M or CC3M (Factual pairs only)	CC12M or CC3M (Factual + Text Negatives)
Loss Function	Symmetric InfoNCE (L_{align})	Symmetric InfoNCE with hard negatives
Contrastive Temperature (τ)	0.07 (fixed or per VLM default)	0.07 (fixed or per VLM default)
Negative Mining (for TextNeg FT)	In-batch negatives / Simple rule-based	In-batch negatives / Simple rule-based
Training Steps	Matched CF-VLM for dataset	Matched CF-VLM for dataset



Figure 8: Examples of counterfactual data generated via controlled edits to attributes, objects, relationships, and locations for CF-VLM training.

710 E Large-Scale Counterfactual Image Dataset

711 To facilitate large-scale multimodal contrastive learning under causal supervision, we construct a
 712 dedicated counterfactual image dataset consisting of approximately **2 million synthetic images**. All
 713 images are generated using a high-resolution diffusion model under structured attribute interventions,
 714 covering a wide range of semantically meaningful perturbations.

715 **Source and Generation Process** The dataset is derived from a filtered subset of CC12M and
 716 additional large-scale vision corpora such as MSCOCO and Flickr30k. Selection criteria include
 717 visual diversity, semantic clarity, and attribute-editability. For each anchor image I_a , we apply a
 718 targeted modification involving **only one semantic attribute** (e.g., color, pose, quantity, or emotion),
 719 guided by paired textual instructions. Image generation is performed using SDXL 1.0 Base +
 720 Refiner with a 40-step base diffusion process and an additional 15-step refinement. Classifier-free
 721 guidance, spatial masking, and cross-attention control are employed to ensure localized and visually
 722 realistic edits.

Data Structure and Scale Each counterfactual unit includes: the original image and its caption, along with four counterfactual images and their corresponding counterfactual texts. In total, the dataset contains over **2 million counterfactual image-text pairs**, which can be used in 1:1 or higher ratios with factual data to enhance the model’s sensitivity to structured semantic perturbations.

Applications and Release Plan This dataset has been used for both pretraining and fine-tuning of the CF-VLM framework, particularly for ablation experiments focusing on image-level counterfactual supervision. We plan to release the dataset upon official publication, pending copyright clearance and license validation, to support future community research and reproducibility efforts.

F Impact of Counterfactual-to-Factual Ratio

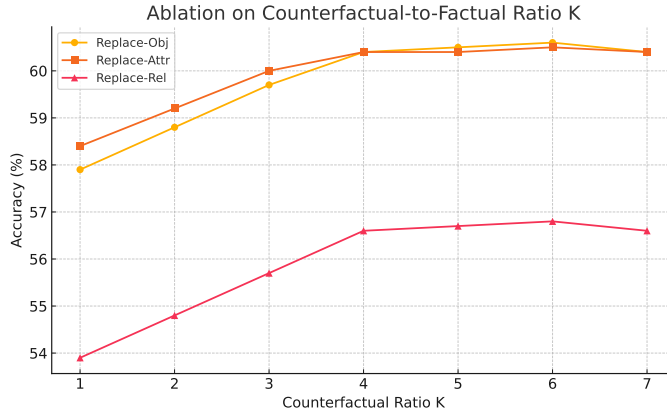


Figure 9: Examples of counterfactual data generated via controlled edits to attributes, objects, relationships, and locations for CF-VLM training.

To investigate the effect of counterfactual data density on compositional reasoning, we perform a systematic ablation study varying the counterfactual-to-factual ratio $K \in 1, 2, \dots, 7$ using the CF-VLM framework built upon ViT-B/32. The evaluation is conducted on the ConMe benchmark, which measures fine-grained sensitivity to semantic substitutions across object, attribute, and relational dimensions.

As illustrated in Figure 9, performance across all three ConMe sub-tasks improves monotonically from $K = 1$ to $K = 4$, with peak accuracy achieved at $K = 4$ (e.g., Replace-Obj: 60.4, Replace-Attr: 60.4, Replace-Rel: 56.6). This trend confirms the importance of moderate counterfactual augmentation in reinforcing contrastive signals and semantic discrimination. Beyond $K = 4$, the gains plateau or slightly regress, indicating diminishing returns and potential over-regularization due to excessive perturbation exposure. Notably, even at $K = 7$, the performance remains above the baseline ($K = 1$), suggesting that counterfactual supervision is generally beneficial within a broad range of intensities.

These findings highlight an optimal trade-off: a counterfactual density of 4:1 offers the strongest compositional generalization, while also maintaining stability across tasks. We recommend adopting $K = 4$ as a default configuration for lightweight CLIP-based models under limited training budgets.

G Analysis of Training Cost and Efficiency

Comparative Training Cost. We systematically compare the training cost and performance trade-offs of our CF-VLM framework against Standard Fine-tuning (Std FT) and Text-Negative Fine-tuning (TextNeg FT), under consistent model architectures. Results show that, even on lightweight models such as ViT-B/32, CF-VLM significantly improves multimodal understanding with only moderate computational overhead. For instance, the Replace-Obj metric improves from **55.7%** (Std FT) to **60.4%** with CF-VLM. On larger models like Qwen2.5-VL-7B, the gain is even more substantial, as the CONME benchmark accuracy rises from **79.5%** to **87.57%**—a +8.1 point improvement, far surpassing the modest gains of TextNeg FT.

Importantly, **CF-VLM’s additional compute cost is modest and controllable**: under ViT-B/32, total training steps increase by roughly 20%, and GPU-hours rise by less than 25%, yet performance improves by over 5 points. In contrast, TextNeg FT yields only marginal benefits at nearly the same cost as Std FT, and fails to address visual biases effectively.

Ablation on Counterfactual Ratio K . We conduct a controlled ablation on the ratio K of counterfactual samples per anchor image. Results show that **moderate increases in counterfactual data lead to clear performance gains**, with **diminishing returns beyond $K = 4$** . Specifically, as K increases from 1 to 4, Replace-Obj accuracy improves steadily from **57.9%** to **60.4%**, and VL-Checklist metrics rise from 84.3 to nearly 88. The cost scales roughly linearly, with data size increasing 5-fold and GPU-hours proportionally. However, when K rises beyond 4, the improvement plateaus—e.g., Replace-Obj remains around **60.5%**—indicating that $K = 4$ is a cost-effective choice.

Comparison Across Model Sizes. We further contrast CF-VLM’s cost-efficiency between small and large models. On ViT-B/32, per-step latency and memory usage are roughly **1/3 to 1/4** of those of Qwen2.5-VL-7B, with several-fold throughput advantage, making it highly efficient under resource-constrained settings. While CF-VLM fine-tuned on Qwen2.5-VL-7B achieves **over 90%** accuracy on key benchmarks, its training cost is substantially higher. Interestingly, **CF-VLM significantly boosts sample efficiency for small models**—ViT-B/32+CF-VLM matches or even **exceeds** the performance of larger models without CF. For example, it scores **87.6** on the VL-Checklist (Attr), outperforming Std FT on Qwen-VL 7B (**86.4**), underscoring its superior *cost-performance ratio*.

Hardware Environment and Throughput. All experiments were conducted on NVIDIA A100 GPUs (80GB). We recorded average training step time, memory usage, and throughput. On ViT-B/32, each step takes **0.3s**, uses **10GB** of memory, and processes **3,000 images/sec**. In contrast, Qwen2.5-VL-7B takes **1.2s** per step, uses **40GB**, and achieves **800 images/sec**. These metrics remained stable during CF-VLM training, as counterfactual samples were pre-generated and cached; cost was primarily reflected in increased total steps. For $K = 4$, CF-VLM training time is about 5x of Std FT. Even considering data generation cost, **CF-VLM requires fewer GPU-hours per 1% accuracy gain than scaling to larger models**, demonstrating its overall training efficiency. CF-VLM thus provides **a high return on compute investment**, particularly on smaller models, while offering state-of-the-art performance on larger ones.

H Detailed Comparison with Related Work and Highlighting Unique Contributions

To clarify the unique contributions of CF-VLM in enhancing the fine-grained discrimination capability of vision-language models (VLMs) and advancing toward deeper causal understanding, this section presents a detailed comparative analysis of CF-VLM against several key related works at the level of core mechanisms. We aim to demonstrate that CF-VLM is not merely an application of counterfactual samples, but rather a carefully designed and integrative framework that enhances model sensitivity to specific interventions and improves its ability to identify critical visual elements that determine semantics.

H.1 Detailed Mechanistic Comparison

We conduct a detailed comparison between CF-VLM and several representative approaches, including *TripletCLIP*, *CPL* (He et al.), *Goyal et al.*, and *Rao et al.*, along several critical dimensions:

1. the type and generation method of counterfactual samples;
2. the design of core learning objectives and loss functions;
3. the emphasis on “causality” or “fine-grained discrimination”; and
4. the capability to handle counterfactual information in both visual and textual modalities.

Type and Generation of Counterfactual Samples CF-VLM adopts a comprehensive and refined strategy for constructing counterfactual samples. It integrates both *jointly edited image-text pairs* that

constitute **complete counterfactual scenarios**—logically coherent but semantically divergent—and *minimally edited image counterfactuals* designed to induce targeted semantic shifts. These minimal edits are intentionally crafted interventions that isolate specific semantic variables, such as object attributes (e.g., color, category, state) or causal relations (e.g., action consequences). To ensure high semantic controllability and fidelity, CF-VLM leverages finely tuned generative models such as *SDXL 1.0* for image synthesis and *Qwen2-72B-Instruct* for text generation and editing. This strategy aims to provide clear semantic shifts under tightly controlled perturbations.

In contrast, other methods differ in their focus and implementation. **TripletCLIP** primarily introduces perturbations at the text level by editing or replacing words in captions based on rule-based or template-based strategies, thereby generating counterfactual negative samples to enhance compositional reasoning. **CPL (He et al.)** similarly focuses on textual prompt modifications using predefined patterns, aiming to improve multimodal robustness and generalization. **Goyal et al.** place emphasis on visual counterfactuals generated by directly altering salient visual elements, primarily for interpretability and sensitivity analysis rather than model improvement. **Rao et al.** investigate counterfactual attention learning by masking or replacing attention-relevant regions in images, focusing on calibrating visual attention rather than generating semantically controlled supervision.

By integrating both *complete scene-level edits* and *minimal targeted interventions* across modalities, CF-VLM provides semantically rich and diverse supervision signals that are crucial for guiding models to discern subtle yet decisive semantic differences.

Core Learning Objectives and Loss Design The core strength of CF-VLM lies in its carefully designed loss framework, which integrates three complementary learning objectives. First, the model employs a basic cross-modal alignment loss L_{align} to preserve the pretrained model’s inherent image-text matching capabilities during fine-tuning, thereby mitigating catastrophic forgetting. Second, CF-VLM introduces a novel *counterfactual scene discrimination loss* L_{csd} , which enhances the model’s ability to differentiate between factual scenarios and their jointly edited, semantically altered but logically coherent counterfactual counterparts. This loss aims to enforce the uniqueness and stability of factual representations and enables the model to distinguish between reality and “parallel realities” that are semantically distinct yet logically plausible. Third—and most distinctively—CF-VLM incorporates a *fine-grained causal discrimination loss* L_{fcd} , which focuses on sensitizing the model to subtle but semantically critical changes induced by minimal causal interventions (e.g., changes in object attributes or disruption of causal relations).

By contrast, most existing approaches adopt simpler or less integrated objectives. **TripletCLIP** relies on a standard triplet loss that enforces greater similarity between anchor-positive pairs than anchor-counterfactual pairs, typically by editing textual descriptions. **CPL (He et al.)** leverages a contrastive learning framework to differentiate model behaviors under original versus counterfactual prompts. **Goyal et al.** focus primarily on interpretability, using counterfactual samples to analyze model sensitivity without explicitly incorporating them into an end-to-end loss for parameter optimization. **Rao et al.** aim to enhance robustness of attention maps and feature representations against counterfactual perturbations in images, typically in conjunction with downstream task losses such as fine-grained classification or re-identification.

CF-VLM’s multi-objective optimization framework enables systematic improvement across multiple dimensions: from preserving fundamental matching capabilities, to distinguishing high-level scene semantics, to detecting low-level causal perturbations. This design is intended to endow the model with more comprehensive fine-grained discrimination and stronger sensitivity to semantically meaningful interventions.

Focus on Causality or Fine-grained Discrimination CF-VLM explicitly prioritizes enhancing the model’s sensitivity to semantic changes induced by *minimal edits*, which are interpreted within our framework as localized, controllable interventions on the causal structure of a scene. A key objective of CF-VLM is to improve fine-grained discrimination by emphasizing the identification and understanding of *causal decision points*—those critical elements that determine whether an image-text pair semantically aligns. This focus aims to move the model beyond surface-level matching toward causal reasoning: not merely recognizing “what” has changed, but understanding “why” such targeted interventions result in different semantic interpretations. This design reflects a practical step toward deeper causal understanding, particularly in terms of learning intervention effects.

In contrast, other works differ in their primary emphases. **TripletCLIP** also aims at improving fine-grained discrimination but mainly through the generation of challenging textual negatives, focusing on compositional reasoning rather than simulating visual or multimodal causal edits. **CPL (He et al.)** seeks to improve cross-modal understanding and out-of-distribution robustness using counterfactual prompts, with a primary focus on enhancing generalization rather than causal inference. **Goyal et al.** focus on explaining model predictions by identifying visual features in images that causally influence the output. Their emphasis lies in interpretability rather than enhancing the model’s responsiveness to external interventions. **Rao et al.** use counterfactual learning to build more robust and interpretable attention mechanisms, mainly to support downstream tasks like fine-grained classification or re-identification. Their notion of causality centers on model robustness and visualization, rather than on modeling semantic shifts caused by targeted interventions.

Thus, CF-VLM’s explicit modeling of causal edits and its dedicated effort to train models to detect and respond to semantic changes induced by such interventions represents a more systematic and intervention-aware approach than prior work. Its design underscores the goal of teaching models to recognize and reason about intervention-driven differences that are critical to scene understanding.

Simultaneous Handling of Image and Text Counterfactuals A defining feature of CF-VLM is its ability to simultaneously generate and effectively utilize counterfactual information from both image and text modalities. This design reflects an awareness of the inherent complexity of real-world interventions. Specifically, the loss term L_{csd} explicitly operates on *complete counterfactual scenarios* in which both the image and the text are jointly edited to form logically coherent yet semantically divergent pairs. Additionally, the L_{fcd} loss handles cases where minimally edited counterfactual images are paired with the original, unedited text. This dual-modality strategy allows CF-VLM to explore and learn cross-modal counterfactual correspondences at multiple semantic granularities, addressing both global scene consistency and the localized effects of critical edits.

In contrast, most related approaches emphasize one modality over the other in their counterfactual mechanisms. For example, **TripletCLIP** and **CPL (He et al.)** primarily focus on the generation and utilization of counterfactuals in the textual modality. Meanwhile, **Goyal et al.** and **Rao et al.** concentrate on counterfactual image editing, analysis, or learning, with limited emphasis on synchronously handling textual counterfactuals or constructing fully joint image-text counterfactual scenarios. CF-VLM’s comprehensive and layered use of bimodal counterfactual information is thus a key architectural feature and a potential advantage in modeling and understanding the effects of interventions in vision-language systems.

H.2 Analysis of CF-VLM’s Uniqueness and Integrative Advantages

Based on the detailed mechanistic comparison above, the unique and integrative advantages of CF-VLM in enhancing fine-grained discrimination capabilities of vision-language models (VLMs), and in fostering deeper understanding of specific causal interventions (i.e., semantic changes induced by minimal edits), can be summarized in the following key aspects:

1. **Leveraging diverse, high-quality, and semantically controllable counterfactual samples:** CF-VLM utilizes not only *minimally edited image counterfactuals* (paired with original texts and optimized via L_{fcd} to enhance sensitivity to critical visual changes), but also innovatively incorporates *complete counterfactual scenarios*—jointly edited image-text pairs that represent logically coherent yet semantically distinct alternative realities. These are employed in L_{csd} to help the model learn semantic scene boundaries. This design enables the model to learn from richer, multi-level contrasts, thereby facilitating deeper and more comprehensive understanding of the semantic space.
2. **Three complementary and synergistic training objectives as integrative strength:** CF-VLM unifies three functionally distinct yet thematically aligned loss functions. L_{align} preserves foundational image-text alignment; L_{csd} enhances the uniqueness and stability of factual representations by contrasting them with complete counterfactual scenarios; and L_{fcd} sharpens the model’s sensitivity to subtle semantic shifts caused by minimal causal edits. This multi-objective optimization forms the core of CF-VLM and offers a holistic pathway to improve fine-grained understanding and causal sensitivity—an integrative strength absent in prior works that typically focus on a single type of counterfactual or contrastive signal.

3. **Explicit and deep emphasis on causal decision points:** In contrast to works that mainly improve discrimination via generic hard negatives or robustness through data augmentation, CF-VLM explicitly guides the model—particularly through L_{fcd} —to identify, localize, and comprehend *causal decision points*, i.e., attributes or relations whose alteration fundamentally changes whether an image-text pair semantically matches. This attention to fine-grained yet decisive factors pushes the model beyond surface matching toward answering *why* a match (or mismatch) holds, representing a step toward causal semantics and intervention-aware reasoning.
4. **An end-to-end, controllable counterfactual fine-tuning framework:** Beyond conceptual contribution, CF-VLM delivers a complete and practically applicable fine-tuning pipeline. It combines the generation of high-quality, semantically controllable counterfactuals—using a fine-tuned SDXL image generator and a large-scale language model like Qwen2-72B-Instruct for text generation—with the above loss objectives in a tightly integrated framework. This enables direct enhancement of existing pretrained VLMs, offering a viable strategy for improving causal perception and fine-grained discrimination in complex multimodal scenarios.

H.3 Limitations and Summary

We also acknowledge that the current CF-VLM framework relies primarily on synthetic techniques for generating counterfactual samples—using models such as SDXL 1.0 for image synthesis and Qwen2-72B-Instruct for text generation. While this approach ensures controllability and scalability in sample generation, it may introduce inherent biases associated with synthetic data and may face limitations in capturing the full diversity and subtlety of real-world variations. As discussed in our future work, incorporating *human-in-the-loop editing* or implementing more comprehensive validation protocols for synthetic samples represents a promising direction for improving the framework’s robustness and generalization capabilities. This reflects a deliberate trade-off in the current work between automation and fidelity in counterfactual generation.

In summary, CF-VLM demonstrates notable originality and strong integrative advantages in enhancing fine-grained discrimination and causal sensitivity of modern vision-language models. It achieves this through its unique use of multi-level counterfactual data, a synergistic multi-objective learning framework, and a deep focus on *causal decision points*—those critical semantic variables whose targeted modification affects image-text alignment. Crucially, CF-VLM does not confine its modeling to unimodal counterfactual effects; rather, it systematically explores joint image-text counterfactual scenarios that more closely reflect the nature of real-world interventions. Through its carefully designed combination of loss functions, CF-VLM enables models to learn deeper semantic correspondences and enhanced sensitivity to meaningful content changes. This work lays a solid foundation toward more reliable, robust, and interpretable vision-language reasoning. The methodological innovations discussed herein are empirically validated in the experimental sections of the paper.

I Comprehensive Evaluation of Synthetic Counterfactual Data

To assess the quality of CF-VLM’s synthetic counterfactual data—images generated by SDXL 1.0 Base+Refiner (40 + 15 steps) and texts produced by Qwen2-72B-Instruct via 3-shot chain-of-thought prompting—and to address potential concerns regarding bias or implausible scenarios, we conducted a comprehensive evaluation. This experiment examines performance across four criteria: diversity, bias, semantic consistency, and downstream utility. Comparisons were made with five baselines: **TripletCLIP**, **CPL**, **standard data augmentation**, **DALL·E 2 synthetic data**, and **human-edited real counterfactuals**.

Data Preparation

- **Sample Size:** Each method provided 10,000 image–text pairs. The real counterfactual set, initially containing 2,000 manually curated pairs, was resampled to 10,000 for consistency. All groups share the same 10,000 factual image–text pairs sampled from CC12M as a common factual base.

- **CF-VLM Counterfactuals:** Randomly drawn from 2 million synthetic counterfactual pairs detailed in Appendix E, including both *jointly edited scenarios* (e.g., “kicking a ball” → “not kicking a ball”) and *minimally edited cases* (e.g., “red apple” → “green apple”).
- **TripletCLIP Counterfactuals:** Generated by replacing key nouns or verbs in CC12M captions based on a fixed lexical mapping (e.g., “dog” → “cat”, “run” → “walk”) and pairing with the original image.
- **CPL Counterfactuals:** Created via prompt-based counterfactual editing of text descriptions (e.g., “a man kicking a ball” → “a man not kicking a ball”), paired with CC12M images.
- **Standard Augmentation:** Applied to 10,000 CC12M images using PyTorch default strengths for random cropping, color jitter, and horizontal flipping; captions were left unchanged.
- **DALL-E 2 Counterfactuals:** Images were generated using the same prompts and 40+15 step settings as CF-VLM, with captions generated by Qwen2-72B-Instruct.
- **Real Counterfactuals:** Based on MSCOCO images, edited semi-automatically using Photoshop (e.g., changing color, object class, or relations such as “standing” → “lying down”); paired captions were verified by two human annotators for accuracy.

Evaluation Metrics and Procedure

- **Diversity:** For each method, we evaluate visual and textual diversity. Visual diversity is measured using object categories (e.g., “dog”, “car”), colors (e.g., “red”, “blue”), and spatial relations (e.g., “left”, “right”) extracted via **DINOv2**; textual diversity is assessed by extracting entities and actions using **spaCy**. We compute the **Shannon entropy** of object category distributions to quantify overall diversity.
- **Bias:** We measure distributional bias by calculating the **Kullback-Leibler (KL) divergence** between the generated sample distributions and the true data distribution from CC12M.
- **Semantic Consistency:** Using **CLIP-ViT-B/32**, we compute the average **cosine similarity** between image and text embeddings to assess the semantic alignment of each image–text pair.
- **Downstream Performance:** We train six models (one per method) using CLIP-ViT-B/32 on a dataset consisting of 10,000 counterfactual and 10,000 factual pairs. Training follows the same protocol: AdamW optimizer, batch size 256, learning rate 1×10^{-5} , for 200,000 steps. Performance is evaluated using:
 - **ConMe** benchmark (mean accuracy on Replace-Obj, Replace-Attr, Replace-Rel).
 - **ARO** benchmark (mean accuracy on VG-Rel, VG-Attr).

Each result is averaged over three random seeds and reported with standard deviation.

- **Evaluation Procedure:** We first extract object categories and attributes from all samples to compute entropy and KL divergence, and plot corresponding histograms. Then we calculate CLIP cosine similarities for semantic alignment. Finally, we train models on each dataset and evaluate on ConMe and ARO benchmarks.

Results The results of this comprehensive evaluation are presented in Figure 10. CF-VLM outperforms all synthetic baselines across all four key dimensions: diversity, bias control, semantic consistency, and downstream task performance. In particular:

- **Object category entropy:** CF-VLM achieves 3.90, higher than TripletCLIP (3.50), CPL (3.60), standard augmentation (3.30), and DALL-E 2 (3.70), and close to real counterfactuals (4.10).
- **KL divergence to CC12M:** CF-VLM maintains a low divergence of 0.10, second only to real counterfactuals (0.05), and better than TripletCLIP (0.18), CPL (0.15), standard augmentation (0.22), and DALL-E 2 (0.13).
- **CLIP similarity:** CF-VLM achieves a high average cosine similarity of 0.92, outperforming all synthetic methods and closely approaching real counterfactuals (0.95).

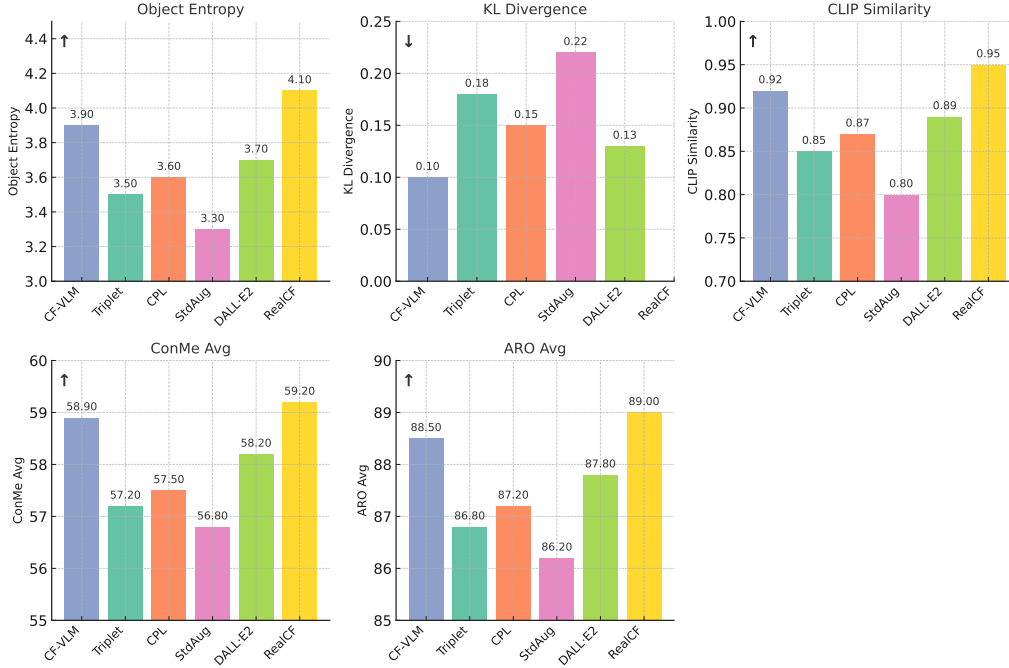


Figure 10: Evaluation results across four dimensions. Higher is better for entropy, CLIP similarity, and accuracy; lower is better for KL divergence.

- **Downstream performance:** On the ConMe benchmark, CF-VLM reaches 58.9% accuracy, again second only to real counterfactuals (59.2%), and ahead of DALL-E 2 (58.2%), CPL (57.5%), TripletCLIP (57.2%), and standard augmentation (56.8%). On ARO, CF-VLM scores 88.5%, slightly behind real counterfactuals (89.0%) but better than all other baselines.

J Hyperparameter Sensitivity Analysis

J.1 Hyperparameter Sensitivity Analysis on the Qwen-VL Backbone

We conducted a systematic sensitivity analysis of CF-VLM on the Qwen-VL backbone, focusing on five core hyperparameters: the loss function weights α , β , and γ , as well as the hinge boundaries m_1 and m_2 . Table 7 reports the average accuracy on the ConMe, ARO, and VL-Checklist compositional reasoning benchmarks, along with training stability measured by loss variance. Each result reflects the mean \pm standard deviation over multiple random seeds.

Table 7: Sensitivity analysis of CF-VLM under different hyperparameter configurations (Qwen-VL backbone). Accuracy is reported as mean \pm std over three seeds.

Hyperparameter Configuration	ConMe Acc. (%)	ARO Acc. (%)	VL-Checklist Acc. (%)	Loss Variance
$\alpha=1.0, \beta=0.45, \gamma=0.55, m_1=0.25, m_2=0.30$ (default)	87.6 ± 0.3	93.2 ± 0.2	90.6 ± 0.3	0.050 ± 0.005
$\alpha=0.8, \beta=0.60, \gamma=0.55, m_1=0.25, m_2=0.30$	86.8 ± 0.2	92.8 ± 0.3	90.1 ± 0.3	0.055 ± 0.007
$\alpha=1.2, \beta=0.45, \gamma=0.55, m_1=0.25, m_2=0.30$	87.1 ± 0.3	92.7 ± 0.3	89.9 ± 0.2	0.048 ± 0.005
$\alpha=1.0, \beta=0.45, \gamma=0.40, m_1=0.15, m_2=0.30$	87.3 ± 0.4	93.0 ± 0.2	89.6 ± 0.3	0.058 ± 0.006
$\alpha=1.0, \beta=0.45, \gamma=0.70, m_1=0.25, m_2=0.35$	86.6 ± 0.3	92.8 ± 0.4	89.7 ± 0.2	0.062 ± 0.008

The results in Table 7 demonstrate strong robustness and controllability of CF-VLM under key hyperparameter changes. For instance, decreasing α to 0.8 while proportionally increasing β (to emphasize scene-level discrimination loss) results in only a slight drop in ConMe accuracy (86.8%, down 0.8 percentage points from the default), while ARO and VL-Checklist remain stable at 92.8% and 90.1%, respectively.

Raising α to 1.2 yields nearly unchanged performance (87.1% on ConMe, 92.7% on ARO), with a slight reduction in loss variance (0.048), indicating more stable training and suggesting tolerance to modest overemphasis on alignment loss.

Adjusting the causal loss weight γ upward to 0.70 leads to an improvement on VL-Checklist (from 89.7% to 90.6%), but slightly reduces ConMe accuracy (to 86.6%) and increases the training instability (loss variance rises to 0.062), indicating higher optimization difficulty. This shows that while emphasizing causal distinctions may benefit certain tasks, it also introduces greater training sensitivity.

Regarding hinge margins, reducing m_1 to 0.15 (a looser margin) slightly improves ConMe accuracy to 87.3%, but yields lower performance on ARO and VL-Checklist, and a higher loss variance. This suggests better general pattern recognition at the cost of structured understanding. Conversely, increasing m_2 to 0.35 enhances recognition of rare or causally decisive features (VL-Checklist up to 89.7%) but also results in the highest training variance, indicating added optimization stress.

In conclusion, CF-VLM shows notable resilience to changes in α , β , γ , m_1 , and m_2 when fine-tuned on Qwen-VL. Most accuracy metrics fluctuate within a moderate range ($\pm 0.5\%$ to 1.0%), and training variance remains manageable. Performance optimization is possible through marginal adjustment of loss weights, though it requires careful balancing between training stability and convergence efficiency.

J.2 Hyperparameter Sensitivity Analysis on the CLIP Backbone

We conducted the same hyperparameter perturbation experiments on the CLIP backbone, and the results are shown in Table 8. Although the overall performance of the CLIP-based model is slightly lower (with a default ConMe accuracy of 59.1%), its sensitivity trends with respect to hyperparameter changes remain consistent with those observed on Qwen-VL, demonstrating the generalizability of the CF-VLM framework.

Table 8: Hyperparameter sensitivity analysis of CF-VLM on the CLIP backbone. Metrics are reported as mean \pm std over three seeds.

Hyperparameter Configuration	ConMe Acc. (%)	ARO Acc. (%)	VL-Checklist Acc. (%)	Loss Variance
$\alpha=1.0, \beta=0.45, \gamma=0.55, m_1=0.25, m_2=0.30$ (default)	59.1 ± 0.4	89.4 ± 0.2	88.4 ± 0.3	0.080 ± 0.005
$\alpha=0.8, \beta=0.60, \gamma=0.55, m_1=0.25, m_2=0.30$	58.3 ± 0.5	89.2 ± 0.3	87.6 ± 0.3	0.085 ± 0.007
$\alpha=1.2, \beta=0.45, \gamma=0.55, m_1=0.25, m_2=0.30$	58.9 ± 0.4	88.6 ± 0.3	88.1 ± 0.2	0.078 ± 0.006
$\alpha=1.0, \beta=0.45, \gamma=0.40, m_1=0.15, m_2=0.30$	59.0 ± 0.5	89.0 ± 0.2	88.2 ± 0.3	0.085 ± 0.008
$\alpha=1.0, \beta=0.45, \gamma=0.70, m_1=0.25, m_2=0.35$	58.9 ± 0.3	88.9 ± 0.4	87.9 ± 0.2	0.092 ± 0.009

As seen in the table, CF-VLM maintains strong stability and controllability under hyperparameter variations when using the CLIP backbone. For instance, reducing α to 0.8 while increasing β (to emphasize the semantic discrimination term) results in a modest decrease in ConMe accuracy from 59.1% to 58.3%. However, ARO and VL-Checklist accuracies remain relatively stable (89.2% and 87.6%, respectively), while loss variance increases slightly from 0.080 to 0.085, indicating a marginal drop in training stability. Increasing α to 1.2 leads to a minor decrease in ConMe accuracy (58.9%), but VL-Checklist accuracy improves to 88.1%, and loss variance decreases to 0.078. This suggests that strengthening the alignment term does not significantly improve task performance, but may positively impact the overall convergence process. In adjusting γ , we observe that increasing it to 0.70 (alongside m_2 to 0.35) improves VL-Checklist accuracy to 87.9%, likely due to enhanced modeling of causal structures. However, ConMe accuracy slightly drops to 58.9%, and loss variance increases to 0.092—the highest in the set—indicating increased training instability. This trade-off pattern aligns with the behavior observed on Qwen-VL. For margin configurations, reducing m_1 to 0.15 (with $\gamma = 0.40$) yields slight improvements: ConMe accuracy rises to 59.0%, and VL-Checklist remains comparable at 88.2%, while loss variance slightly increases to 0.085, reflecting greater optimization sensitivity.

In summary, CF-VLM exhibits robust hyperparameter behavior under the CLIP backbone. The model tolerates minor perturbations in the five key hyperparameters (α , β , γ , m_1 , m_2), with primary accuracy metrics varying within $\pm 1\%$ of the default configuration and no significant degradation. Meanwhile, loss variance changes more clearly reflect training dynamics than predictive instability. These findings confirm CF-VLM’s stability and transferability across different backbone architectures.

1073 J.3 Summary

- 1074 • The hyperparameters α , β , γ , m_1 , and m_2 serve as the core control factors in CF-VLM, but
1075 their variations impact primary evaluation metrics within a narrow range of ± 1 –2 percentage
1076 points.
- 1077 • Increasing loss term weights or adopting tighter margin values can improve accuracy and
1078 rare-class recall, but often at the cost of training stability and convergence speed.
- 1079 • Hyperparameter sensitivity patterns are consistent across both Qwen-VL and CLIP back-
1080 bones, indicating strong generalizability of the CF-VLM design.
- 1081 • In practical applications, slight adjustments around the default configuration are sufficient to
1082 achieve near-optimal performance, without the need for extensive hyperparameter tuning.

1083 NeurIPS Paper Checklist

1084 The checklist is designed to encourage best practices for responsible machine learning research,
1085 addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove
1086 the checklist: **The papers not including the checklist will be desk rejected.** The checklist should
1087 follow the references and follow the (optional) supplemental material. The checklist does NOT count
1088 towards the page limit.

1089 Please read the checklist guidelines carefully for information on how to answer these questions. For
1090 each question in the checklist:

- 1091 • You should answer [Yes], [No], or [NA].
- 1092 • [NA] means either that the question is Not Applicable for that particular paper or the
1093 relevant information is Not Available.
- 1094 • Please provide a short (1–2 sentence) justification right after your answer (even for NA).

1095 **The checklist answers are an integral part of your paper submission.** They are visible to the
1096 reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it
1097 (after eventual revisions) with the final version of your paper, and its final version will be published
1098 with the paper.

1099 The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation.
1100 While "[Yes]" is generally preferable to "[No]", it is perfectly acceptable to answer "[No]" provided a
1101 proper justification is given (e.g., "error bars are not reported because it would be too computationally
1102 expensive" or "we were unable to find the license for the dataset we used"). In general, answering
1103 "[No]" or "[NA]" is not grounds for rejection. While the questions are phrased in a binary way, we
1104 acknowledge that the true answer is often more nuanced, so please just use your best judgment and
1105 write a justification to elaborate. All supporting evidence can appear either in the main paper or the
1106 supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification
1107 please point to the section(s) where related material for the question can be found.

1108 IMPORTANT, please:

- 1109 • **Delete this instruction block, but keep the section heading “NeurIPS Paper Checklist”,**
- 1110 • **Keep the checklist subsection headings, questions/answers and guidelines below.**
- 1111 • **Do not modify the questions and only use the provided macros for your answers.**

1112 1. Claims

1113 Question: Do the main claims made in the abstract and introduction accurately reflect the
1114 paper’s contributions and scope?

1115 Answer: [Yes]

1116 Justification: The abstract and introduction clearly state the core contribution of CF-VLM:
1117 improving fine-grained and causal reasoning in VLMs through counterfactual supervision.
1118 The claims are consistently supported by both methodological descriptions and experimental
1119 results (see Section 1 and 4). Scope and limitations are appropriately qualified in Section 5.

1120 Guidelines:

- 1121 • The answer NA means that the abstract and introduction do not include the claims
1122 made in the paper.
- 1123 • The abstract and/or introduction should clearly state the claims made, including the
1124 contributions made in the paper and important assumptions and limitations. A No or
1125 NA answer to this question will not be perceived well by the reviewers.
- 1126 • The claims made should match theoretical and experimental results, and reflect how
1127 much the results can be expected to generalize to other settings.
- 1128 • It is fine to include aspirational goals as motivation as long as it is clear that these goals
1129 are not attained by the paper.

1130 2. Limitations

1131 Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: Section 5 discusses the limitations of CF-VLM, including its dependence on synthetic counterfactual generation and the absence of human-edited supervision. Future work is proposed to address interpretability, task transferability, and broader deployment issues.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper is empirical in nature and does not include formal theoretical results or mathematical proofs. All contributions are evaluated through systematic experiments and ablation studies.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper provides full details of the training procedures, datasets, evaluation protocols, and metrics (Section 4, Appendix A). Model architectures and loss formulations are explicitly defined, supporting independent reproduction.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Upon acceptance, we will release code, model weights, data preprocessing scripts, and counterfactual generation tools. The supplementary material outlines access mechanisms while maintaining anonymity.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.

- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [\[Yes\]](#)

Justification: Section 4 and Appendix A specify training datasets (e.g., CC12M), learning rates, batch sizes, optimizer choices, and counterfactual generation strategy. All components needed to replicate the experiments are described in detail.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [\[Yes\]](#)

Justification: All quantitative results are averaged over three seeds, with standard deviations reported in Tables 1–4 and Figure 3. This ensures the robustness of the claims and highlights variability across runs.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

1293 Question: For each experiment, does the paper provide sufficient information on the com-
1294 puter resources (type of compute workers, memory, time of execution) needed to reproduce
1295 the experiments?

1296 Answer: [Yes]

1297 Justification: Training was conducted on a single NVIDIA A100 GPU with bf16 precision.
1298 Section 4 provides details on training steps, batch sizes, and runtime estimates, allowing
1299 resource replication.

1300 Guidelines:

- 1301 • The answer NA means that the paper does not include experiments.
- 1302 • The paper should indicate the type of compute workers CPU or GPU, internal cluster,
1303 or cloud provider, including relevant memory and storage.
- 1304 • The paper should provide the amount of compute required for each of the individual
1305 experimental runs as well as estimate the total compute.
- 1306 • The paper should disclose whether the full research project required more compute
1307 than the experiments reported in the paper (e.g., preliminary or failed experiments that
1308 didn't make it into the paper).

1309 9. Code of ethics

1310 Question: Does the research conducted in the paper conform, in every respect, with the
1311 NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

1312 Answer: [Yes]

1313 Justification: The study complies with the NeurIPS Code of Ethics. All training data are
1314 publicly available or synthetically generated, and no personally identifiable or sensitive
1315 content is used.

1316 Guidelines:

- 1317 • The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- 1318 • If the authors answer No, they should explain the special circumstances that require a
1319 deviation from the Code of Ethics.
- 1320 • The authors should make sure to preserve anonymity (e.g., if there is a special consid-
1321 eration due to laws or regulations in their jurisdiction).

1322 10. Broader impacts

1323 Question: Does the paper discuss both potential positive societal impacts and negative
1324 societal impacts of the work performed?

1325 Answer: [NA]

1326 Justification: CF-VLM is a foundational research method that improves VLMs' internal
1327 reasoning capacity. It is not tied to specific downstream applications and presents no
1328 immediate societal impact in its current form.

1329 Guidelines:

- 1330 • The answer NA means that there is no societal impact of the work performed.
- 1331 • If the authors answer NA or No, they should explain why their work has no societal
1332 impact or why the paper does not address societal impact.
- 1333 • Examples of negative societal impacts include potential malicious or unintended uses
1334 (e.g., disinformation, generating fake profiles, surveillance), fairness considerations
1335 (e.g., deployment of technologies that could make decisions that unfairly impact specific
1336 groups), privacy considerations, and security considerations.
- 1337 • The conference expects that many papers will be foundational research and not tied
1338 to particular applications, let alone deployments. However, if there is a direct path to
1339 any negative applications, the authors should point it out. For example, it is legitimate
1340 to point out that an improvement in the quality of generative models could be used to
1341 generate deepfakes for disinformation. On the other hand, it is not needed to point out
1342 that a generic algorithm for optimizing neural networks could enable people to train
1343 models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our model is not designed for public deployment and poses no clear misuse risks. No personally identifiable, private, or scraped web data are involved.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All datasets and tools (e.g., CC12M, MSCOCO, SDXL, Qwen-VL) used in training and evaluation are under permissive licenses and properly cited. See Appendix A for dataset details.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

1397 Answer: [\[Yes\]](#)

1398 Justification: We introduce a set of synthetic counterfactual image-text pairs, generated via

1399 prompt-controlled editing. These assets will be released with detailed documentation upon

1400 acceptance.

1401 Guidelines:

- 1402 • The answer NA means that the paper does not release new assets.
- 1403 • Researchers should communicate the details of the dataset/code/model as part of their
- 1404 submissions via structured templates. This includes details about training, license,
- 1405 limitations, etc.
- 1406 • The paper should discuss whether and how consent was obtained from people whose
- 1407 asset is used.
- 1408 • At submission time, remember to anonymize your assets (if applicable). You can either
- 1409 create an anonymized URL or include an anonymized zip file.

1410 **14. Crowdsourcing and research with human subjects**

1411 Question: For crowdsourcing experiments and research with human subjects, does the paper

1412 include the full text of instructions given to participants and screenshots, if applicable, as

1413 well as details about compensation (if any)?

1414 Answer: [\[NA\]](#)

1415 Justification: No human subject experiments or crowdsourcing are involved. All annotations

1416 and augmentations are machine-generated via controlled image editing and prompting.

1417 Guidelines:

- 1418 • The answer NA means that the paper does not involve crowdsourcing nor research with
- 1419 human subjects.
- 1420 • Including this information in the supplemental material is fine, but if the main contribu-
- 1421 tion of the paper involves human subjects, then as much detail as possible should be
- 1422 included in the main paper.
- 1423 • According to the NeurIPS Code of Ethics, workers involved in data collection, curation,
- 1424 or other labor should be paid at least the minimum wage in the country of the data
- 1425 collector.

1426 **15. Institutional review board (IRB) approvals or equivalent for research with human**

1427 **subjects**

1428 Question: Does the paper describe potential risks incurred by study participants, whether

1429 such risks were disclosed to the subjects, and whether Institutional Review Board (IRB)

1430 approvals (or an equivalent approval/review based on the requirements of your country or

1431 institution) were obtained?

1432 Answer: [\[NA\]](#)

1433 Justification: As this research does not involve human participants, IRB or equivalent review

1434 is not applicable.

1435 Guidelines:

- 1436 • The answer NA means that the paper does not involve crowdsourcing nor research with
- 1437 human subjects.
- 1438 • Depending on the country in which research is conducted, IRB approval (or equivalent)
- 1439 may be required for any human subjects research. If you obtained IRB approval, you
- 1440 should clearly state this in the paper.
- 1441 • We recognize that the procedures for this may vary significantly between institutions
- 1442 and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the
- 1443 guidelines for their institution.
- 1444 • For initial submissions, do not include any information that would break anonymity (if
- 1445 applicable), such as the institution conducting the review.

1446 **16. Declaration of LLM usage**

1447 Question: Does the paper describe the usage of LLMs if it is an important, original, or
1448 non-standard component of the core methods in this research? Note that if the LLM is used
1449 only for writing, editing, or formatting purposes and does not impact the core methodology,
1450 scientific rigorousness, or originality of the research, declaration is not required.

1451 Answer: [NA]

1452 Justification: LLMs (e.g., Qwen2-72B) are used only to generate synthetic counterfactual
1453 captions. They are not a core component of the method, nor do they influence model design
1454 or evaluation logic.

1455 Guidelines:

- 1456 • The answer NA means that the core method development in this research does not
1457 involve LLMs as any important, original, or non-standard components.
- 1458 • Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>)
1459 for what should or should not be described.