DETECTING FINE-GRAINED DIFFERENCES BETWEEN IMAGE DATASETS

Anonymous authors
Paper under double-blind review

ABSTRACT

Detecting whether and in particular *how* two data distributions differ is a fundamental statistical problem. For example, we might be interested in which concepts (expressed in natural language) differ between two classes, the outputs of different generative models, or the same model with different prompt specifications. When working with large and complex image datasets, we intuitively think that first selecting concepts from a broad list using fine-grained patch embeddings, and then optimizing the concepts to better fit the data is an effective workflow. However, prior work instead uses MLLMs to propose concepts and image-level embeddings to select them. Therefore, we systematically compare these techniques and find that selecting concepts from broad lists can replace MLLM-proposals at obviously lower cost. MLLM inference can then be spent on optimizing selected concepts instead to achieve substantial improvements. Additionally, by designing a new benchmark that focuses on low-prevalence and multi-concept distinctions, we find that selecting nonredundant concepts instead of only the most distinguishing concepts matters, and we introduce techniques for how to effectively do this. Finally, we show how our methods address important problems like detecting spurious correlations and biases. Overall, our work presents a fundamental perspective shift on how we analyze interpretable distributional differences between datasets.

1 INTRODUCTION

Many methods can tell us that two datasets differ, but comparatively few can explain what makes them different in human-interpretable terms. For example, classification separates class-conditional distributions, and distribution-shift detection compares training and deployment distributions. However, most methods tell us only whether or how much two distributions differ. To make this understandable to humans and amenable to analyses, we also need to find out *what* makes them different. *Dataset diffing* addresses this by searching for concepts (expressed in natural language) whose prevalence differs between two datasets. For images, which are the focus of this paper, this requires jointly solving three challenges: perceiving evidence in images, aggregating it across the entire data, and expressing the resulting distinction in language.

Beyond being an interesting problem, dataset diffing has many important and practical applications. For example, dataset diffing methods are used for dataset organization (Liu et al., 2026), temporal trend analysis in city scenes (Deng et al., 2025) and art (Olshausen & DeWeese, 2010; Ginosar et al., 2015), understanding differences between generative models (Jiang et al., 2025; Dunlap et al., 2025), and understanding classifier failure modes (Eyuboglu et al., 2022; Dunlap et al., 2024). Furthermore, dataset diffing can be extended to detecting spurious correlations (Kim et al., 2024; Choi et al., 2025) and biases (D'Incà et al., 2024), which are the focus of our applications.

Our approach to dataset diffing is as follows: We start with a broad and general wordlist that captures the fundamental concepts that exist in language. We then use this list to efficiently select the rough concepts whose prevalence differs between the compared datasets. To detect their prevalence, we use patch embeddings, which function as a zero-shot fine-grained concept detector. Image-level embeddings cannot achieve this, because they often submerge details. Finally, we optionally optimize the concepts using an MLLM that receives sampled images and selected concepts and outputs improved concept candidates. In comparison to existing literature, this is a fundamental shift: We don't derive concepts directly from the data, e.g., by repeatedly proposing them with an MLLM, but triangulate

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

the data with a general list that can test a large number of concepts efficiently, and then optimize them. Prior work proposes a number of concepts and then ranks them, and we first select concepts and then optimize them. In addition, we focus on complex datasets that are distinguished by multiple concepts with individually low prevalence, which requires nonredundant concept selection, because many semantically related concepts tend to score similarly while not adding information and instead suppressing other differences. This focus is also novel in comparison to existing benchmarks that focus on individual distinguishing concepts that are represented in all images.

In summary, our contributions are as follows: (1) We fill an important gap in the literature by systematically demonstrating the advantages and opportunities of fine-grained, patch-level image representations for dataset diffing; (2) We show that (expensive) MLLM proposals can be replaced by taking concepts from a broad, static wordlist, and we propose to spend the MLLM budget on post-selection concept optimization instead. (3) We show that standard benchmarks (VisDiffBench and Metashift) are saturated and propose a more challenging setting where ground truth concepts are sparser and noisier. We show that in this setting, redundancy-aware concept selection and coverage-seeking MLLM optimization are necessary for strong performance, but no method saturates the benchmark; (4) We demonstrate through two relevant applications (finding spurious correlations and demographic biases in text-to-image models) that patch-level embeddings and dataset diffing together enable insights at scale that are hard to achieve with image-level embeddings alone. Hence, this paper is a relevant contribution to the community with high practical value.

2 Related Work

Dataset Diffing in Image Datasets. Existing approaches for image dataset diffing can be categorized into two main strategies. The first approach uses external models to convert each image into a bag-of-visual-words representation. This process extracts objects and attributes in an image through captioning (Kim et al., 2024; Liu et al., 2026), image tagging (Zhang et al., 2024; Huang et al., 2024), object detection (Zhu et al., 2025), concept predictors (Oikarinen et al., 2023; Konforti et al., 2026), or sparse decomposition of image embeddings (Choi et al., 2025; Jiang et al., 2025). We evaluate captioning and tagging approaches in Section 4 and find that they perform well, but not optimally, and they are costly as they require MLLM inference for a large number of images. Concept-based models with a closed concept set may not include important concepts that are relevant for a dataset, while open concept sets (Oikarinen et al., 2023; Zhu et al., 2022) depend on the underlying multimodal embedding space, which we investigate in this work.

The second approach ("propose-then-rank", PTR) involves sampling small sets of images from the contrasted datasets and asking an MLLM to propose distinguishing features (Dunlap et al., 2024; Liu et al., 2026). We show in Section 4 and Appendix A.4 that PTR is robust to low prevalence of target concepts, but the same performance or better can be achieved using a generic wordlist. Therefore, we propose to spend the expensive MLLM inference on post-selection optimization, which enables higher gains on complex datasets (Section 5). There, selecting diverse concepts becomes important, because taking the top-k highest-ranking concepts suffers from high redundancy.

Dataset Diffing in Text-Only Datasets. Methods for explaining differences in text-only datasets mainly follow the "propose-then-rank" approach described above (Wang et al., 2023; Dunlap et al., 2024). Initially, based on a set of samples, a language model proposes potential explanations, which are sometimes clustered (Findeis et al., 2024; Bravansky et al., 2025). These proposed explanations are then evaluated against the documents to determine which documents are associated with each candidate explanation. Finally, a suitable subset of candidates is selected based on the specific goal, such as clustering (Wang et al., 2023) or "inverse constitutional AI" (Findeis et al., 2024). Zhong et al. (2024) propose an additional iterative process to identify better candidate concepts. We show that a similar idea for images works well and enables substantial improvements (Sections 4 and 5).

Explaining Distribution Shifts. Dataset diffing can also be viewed as characterizing distribution shifts, as both tasks assume that datasets differ along a few well-defined attributes. Most work on explaining distribution shifts uses optimal transport (OT). For instance, Kulinski & Inouye (2023) use OT to identify a low-dimensional shift direction in multimodal image embedding space. Then, we can interpret the identified direction by retrieving aligned text concepts. Naumann et al. (2026) propose a similar method but find multiple subspaces. Other work finds a few sample images that best describe the shift, like the most and least shifted samples in OT matchings (Brockmeier et al.,

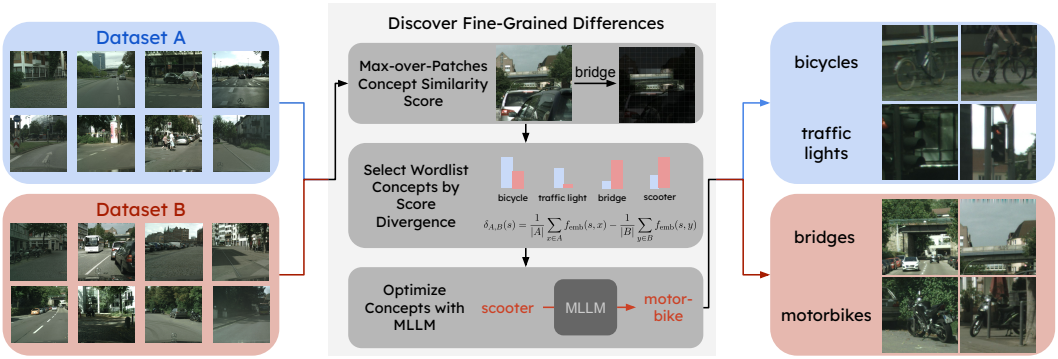


Figure 1: **How we detect fine-grained differences between datasets.** Left: We are given two sets A and B consisting of natural images. Center: We compute multimodal patch embeddings and calculate max-over-patches cosine similarities to corresponding concept text encodings. Then, we find diverse concepts whose score distribution differs between the sets. Finally, we use an MLLM to refine concepts. Right: The method returns a list of distinguishing concepts for each set.

2021; Hulkund et al., 2022) or through image translation (Kulinski & Inouye, 2022; 2023). Our experiments in Section 4 show that subspace-based approaches yield comparatively weak performance, because interpreting subspaces in multimodal embedding spaces is difficult and can introduce noise. Furthermore, OT methods do not trivially generalize to patch-based embedding spaces, because no clear mapping of patches exists across images.

3 Fine-Grained Dataset Diffing with Multimodal Optimization

The task we address is defined as follows: Given two sets of images A and B, find a list of concepts that together describe the differences between the images in A and the images in B, i.e., a list of concepts that are more prevalent in images in A (but not B), and vice versa. This problem can be formalized by the following optimization problem, where we want to find optimal concepts s^* that maximally distinguish A and B by comparing their prevalence:

$$s^* = \arg\max_s \mathcal{R}(s) = \arg\max_s \mathbb{E}_{x\sim A}[f(s,x)] - \mathbb{E}_{y\sim B}[f(s,y)]. \tag{1}$$

Here, s is a natural language description of a possible dataset difference and f is a function that scores how relevant s is to an image x or y, i.e. higher scores indicate higher relevance. This formulation makes clear how dataset diffing entangles a search problem over language (finding s) with a perception problem (evaluating f).

Fine-Grained vs. Image-Level Embeddings. One main focus of this paper is the perception function f. As f needs to be evaluated many times, for efficiency reasons a vector-similarity-based score is necessary. Concretely, we compare image-level embeddings like SigLIP2 (Tschannen et al., 2025), which have been used in prior work, with patch-level multimodal embeddings like FG-CLIP2 (Xie et al., 2026). In both cases, let ϕ denote the image encoder and ψ the text encoder. We then set

$$f_{\text{emb}}^{\text{CLIP}}(s,x) = \psi(s)^\top \phi(x) \tag{2}$$

$$f_{\text{emb}}^{\text{FG-CLIP}}(s,x) = \max_{i\in\{1,\dots,\rho\}} \psi(s)^\top \phi_i(x). \tag{3}$$

Here, FG-CLIP returns a patch embedding matrix of shape $d \times \rho$, where ρ denotes the number of patches and $\phi_i(x)$ denotes the i-th column.

After calculating the f_{emb} scores for a concept and all images in sets A and B, we calculate a divergence score $\delta_{A,B}(s)$ between the score distributions in A and B. This score indicates how prevalent the concept is in A in contrast to B, and vice versa. Depending on the application, different divergence scores can be chosen (see Appendix D.4). For Section 4, we use the difference of means:

$$\delta_{A,B}(s) = \frac{1}{|A|}\sum_{x\in A} f_{\text{emb}}(s,x) - \frac{1}{|B|}\sum_{y\in B} f_{\text{emb}}(s,y). \tag{4}$$



Figure 2: **Example images for VisDiffBench (left) and MetaShift (right).** The summer (A, top) and autumn (B, bottom) sets differ in a whole-image color shift, whereas the helmet (A, top) and motorcycle (B, bottom) sets differ in small objects in otherwise similar street scenes.

Given M candidate concepts $\mathcal{C} = \{s_1, \ldots, s_M\}$, we select k concepts with large $\delta_{A,B}$ as characterizing A, and k concepts with low $\delta_{A,B}$ as characterizing B. In Section 4, we evaluate combining both embedding spaces, and Section 5 focuses on concept selection strategies in complex datasets.

MLLM-Driven Concept Optimization. The initial pool of concepts may be suboptimal, and the k selected concepts may be noisy. Therefore, we iteratively optimize the pool using an MLLM; i.e., we implement the search for s^* as an MLLM-driven optimization problem, similar to previous approaches that utilize LLMs as optimizers (Yang et al., 2024; Fernando et al., 2024). We provide an MLLM with the $2 \times k$ selected concepts and their scores and then request it to generate new candidates that better describe the difference between A and B. Specifically, we provide the following information to an MLLM: (i) An image displaying a 2×5 grid of two rows of images selected from sets A and B; (ii) For both sets A and B, we provide a list of concepts that f scores highly, irrespective of the difference between the two sets. As an initial concept pool to choose from, we use the stopword-filtered COCA corpus wordlist (Davies, 2010), which allows truncating by frequency.

4 Evaluation on VisDiffBench and MetaShift

In the first set of experiments, we get an understanding of which methods perform well. Our main insights are: (1) Standard benchmarks are saturated, as most methods perform well. (2) As expected, patch embeddings are optimal for detecting differences in smaller attributes and objects, and image-level embeddings are optimal for salient themes present in the entire image. (3) Wordlist-based methods already achieve strong performance while being significantly cheaper than MLLM-based concept proposals. MLLM optimization further improves concepts in this setting.

Datasets. We evaluate all methods on VisDiffBench (Dunlap et al., 2024) and MetaShift (Liang & Zou, 2022). VisDiffBench contains 150 subtasks, each consisting of image set pairs (A and B), which are organized into three difficulty levels (easy, medium, and hard). Each image set in VisDiffBench contains exactly 100 images. For MetaShift, we construct a suitable subset of the full benchmark. First, we generate MetaShift for the recommended classes (cat, dog, bus, truck, elephant, horse, bowl, and cup), which results in 212 subtasks. Next, for each image, we use Gemma4-31B to verify the presence of the distinguishing attribute. Images where Gemma4 does not detect the target attribute are removed. Then, we retain the subtasks where both image sets A and B contain at least 20 images each, and we cap the number of images per set at 100. This yields 68 remaining subtasks, with an average of 52.6 images per set. See Fig. 2 for examples from VisDiffBench and MetaShift.

Evaluation. We evaluate descriptions of differences between datasets A and B using three metrics that cover (1) *discrimination* (Disc.), i.e. how well the returned concepts separate the datasets. This is measured by the accuracy an MLLM achieves in classifying images into sets A and B based on the concepts. (2) *relevance* (Rel.), i.e. whether the concepts actually apply to images in the data. This is measured by the average number of concepts an MLLM attributes to images. (3) *embedding similarity* (Sim.), which measures the maximum cosine embedding similarity between the ground truth concept and any proposed concept. All three metrics are based on $k = 10$ concepts returned per side (A and B) by each method. The MLLM used for Disc. and Rel. is Gemma4-31B (Table 1), and we include results for Qwen3.8-27B in Appendix A.1 alongside additional details on the metrics.

	VDB-Easy			VDB-Med.			VDB-Hard			MetaShift		
	D	Rel	Sim.	D	Rel	Sim.	D	Rel	Sim.	D	Rel	Sim.
MLLM + SigLIP (PTR)	0.981	0.809	**0.549**	0.891	0.873	0.425	0.815	0.899	0.331	0.891	0.601	0.441
MLLM + FG-CLIP (PTR)	**0.983**	0.853	0.523	**0.897**	0.907	0.424	0.816	0.940	0.353	0.884	0.663	0.473
Distrib. Shift: U-WaX	0.972	0.335	0.306	0.816	0.262	0.185	0.757	0.196	0.147	0.834	0.232	0.400
Distrib. Shift: Linear	0.969	0.347	0.319	0.842	0.307	0.200	0.752	0.221	0.157	0.844	0.239	0.405
Caption Summary	0.974	0.613	0.338	0.858	0.734	0.267	0.805	0.654	0.236	0.825	0.423	0.252
Image Tagging: MLLM	0.971	0.585	0.346	0.866	0.666	0.288	0.800	0.639	0.243	0.818	0.412	0.310
Image Tagging: RAM++	0.975	**0.907**	0.335	0.833	**0.908**	0.262	0.719	**0.969**	0.219	0.854	**0.767**	0.424
COCA + FG-CLIP	0.963	0.548	0.283	0.844	0.532	0.203	0.739	0.458	0.134	**0.904**	0.510	**0.501**
COCA + SigLIP	0.976	0.679	0.336	0.846	0.672	0.275	0.792	0.680	0.275	0.738	0.402	0.306
+ MLLM Optim.	0.982	0.777	0.511	0.875	0.757	**0.445**	**0.830**	0.854	**0.402**	0.757	0.408	0.307
+ FG-CLIP	0.982	0.746	0.489	0.888	0.784	0.417	0.817	0.743	0.341	0.884	0.546	0.465
True Concept	0.982	-	-	0.889	-	-	0.843	-	-	0.932	-	-

Table 1: **Performance comparison on VisDiffBench and MetaShift.** We compare the methods across three metrics: discrimination (D), relevance (Rel), and embedding similarity (Sim). Best scores per benchmark and metric are highlighted in bold, and second-best scores are underlined.

Baselines. We evaluate the following baselines: (1) propose-then-rank, (2) distribution shift explanation approaches, and (3) tagging + aggregation methods. For (1) *propose-then-rank*, we follow Dunlap et al. (2024), who use an MLLM to propose candidate concepts from $2 \times k$ (we use $k = 5$) image samples from set A and B. We then combine all 10 samples in one composite image and ask an MLLM to propose 10 distinguishing concepts per side, aggregate them over 20 iterations, and rank them with SigLIP or FG-CLIP text-image similarity. See Appendix D.1 for details.

For (2) *distribution shift explanations*, we detect subspaces in the SigLIP2 image embedding space and find text concepts that are aligned with the identified subspaces. We evaluate two approaches: U-WaX (Naumann et al., 2026), which uses optimal transport to find subspaces, and alternatively, we learn orthogonal linear directions from linear support vector classifiers to distinguish between the embeddings of images in both sets. Details are in Appendix D.2. For (3) *tagging + aggregation methods*, we extract relevant textual information from each image using a captioning model and then use an LLM to aggregate the resulting information into candidate concepts. We evaluate three tagging variations: (a) generating a long-form caption for each image; (b) extracting a structured list of objects and their attributes; and (c) extracting visual concepts using the RAM++ model (Zhang et al., 2024). Because RAM++ operates on a fixed label space, here we directly select the 10 concepts with the largest count difference between sets A and B. See Appendix D.3 for details.

Methods. We evaluate our proposed embedding-based approach (Section 3), which scores the stopword-filtered COCA wordlist (M=54,000 concepts) against all images using either SigLIP image-level or FG-CLIP patch-level embeddings and selects the k=10 concepts per side with the largest difference-of-means margin (Eq. (4)). To refine the initial selection, we optionally apply MLLM-driven concept optimization (Section 3), which iteratively proposes and evaluates new candidate phrases given visual examples and the current concept pool. We further evaluate a hybrid configuration that combines both embedding spaces by taking the maximum of their calibrated divergence scores (see Appendix D.4 for technical details).

Results. As shown in Table 1, image-level embeddings (SigLIP) work well on VDB, where distinguishing concepts are visually salient (VDB-Hard: $D = 0.792$ vs. oracle $D = 0.843$), but fail on localized concepts in MetaShift ($D = 0.738$ vs. oracle $D = 0.932$; see Appendix A.2 for an ablation). Conversely, patch-level embeddings (FG-CLIP) work well on MetaShift ($D = 0.904$) but less well on abstract concepts in VDB (also see Appendix A.3 for a test of FG-CLIP fine-grained classification). MLLM-based concept optimization improves SigLIP on VDB-Hard (Rec from 0.275 $\rightarrow 0.402$), closing the gap to the ground truth. Our hybrid approach (COCA + MLLM Optim. + FG-CLIP) balances both representations and achieves strong performance across both benchmarks.

Tagging methods (e.g., RAM++) achieve reasonable discrimination performance, which confirms the need for localized detection, but MLLM-based captioning per image is prohibitively expensive. Distribution shift baselines (U-WaX, Linear) consistently underperform (Sim. ≤ 0.200), which indicates that interpreting geometric shifts in language yields noisy concepts even in multimodal spaces.

Finally, PTR performs strongly overall ($D = 0.891$ on MetaShift). On the one hand, its success reveals a fundamental benchmark limitation: shifts are so isolated and consistent that observing

the entire distribution is unnecessary, and we only need to look at a few images. On the other hand, in Appendix A.4 we show that PTR reliably detects even low-prevalence concepts. So the actual question is about cost. Here, we find that wordlist-based methods achieve very similar performance without any MLLM inference, and surpass MLLM proposals when inference is spent on post-selection optimization. However, real-world datasets are more complex than the single-target, high-prevalence shifts modeled in VisDiffBench and MetaShift. Therefore, we design an experiment to explicitly test methods in noisy and complex datasets.

5 Evaluation on Noisy and Multi-Target Datasets

Setting and Dataset Construction. The results in Section 4 above show that, depending on the dataset, different embedding models are necessary to obtain optimal performance. In particular, in real datasets, generally more than one concept determines the difference, and concepts may not be exclusively tied to one set. Therefore, we construct a harder setting that involves multiple concepts. Given a dataset that has image-level concept annotations, i.e., indicators of which concepts are present in the image, we assign a role for each concept. A concept can be aligned to Set A, aligned to Set B, or neutral. For aligned concepts, between 75% and 95% of images with the concept are in the respective set, and for neutral concepts, at most 60% of images are in either set. Every class is additionally assigned an overall prevalence that determines how frequent the concept is in the entire dataset (A and B combined). This construction directly addresses the limitations we mentioned earlier: It includes multiple concepts per side, and even for aligned concepts, a relevant portion can still be present in the other set. For technical details on the dataset sampling, see Appendix E.1.

Evaluation. In Section 4, we have shown that methods find discriminative and relevant concepts. However, in the complex and noisy datasets that we evaluate in this section, due to the diversity of the target concepts, both are less useful, and even ground-truth concepts don't discriminate sets A and B well in the classification sense (see Appendix E.4 and Table 9).

Instead, we focus on recall, i.e., whether methods can recover the ground-truth-aligned concepts. We measure this from three perspectives: (1) *MCC-R* measures how well each ground-truth concept is recovered by any predicted concept. For this, we calculate the maximum Matthews Correlation Coefficient (MCC) across all possible binarization thresholds t on the image-concept similarity scores. (2) *Sim.* measures the maximum cosine embedding similarity between a ground-truth concept and any predicted concept. (3) R_J uses an LLM to select the ground-truth concepts that are implied by any predicted concept and then calculates recall on this selection.

For MCC, a ground-truth concept is considered recovered if any predicted concept achieves MCC ≥ 0.5. Intuitively, MCC-R measures how well the predicted structure of the data (predicted concepts + scores) that a user would see matches the ground-truth structure induced by ground-truth concepts and binary presence indicators. See Appendix E.4 for details on the metrics.

Methods. For this experiment, we focus on three contrasts: **(1)** How do SigLIP and FG-CLIP compare in complex datasets? **(2)** Can static wordlists compete with MLLM proposals? **(3)** How can we optimally select and improve the concepts? Accordingly, on all datasets, we compare MLLM-proposed concepts (100 times 2×10 proposals, i.e., 10 concepts for each side for 100 2×5 image samples) vs. the static COCA wordlist, SigLIP vs. FG-CLIP f_{emb} similarity scores used to rank concepts, and we show that suitable concept selection strategies yield significant gains. Our main insight is that in this setting, simply selecting the top-10 highly ranked concepts is not optimal, because this leads to high redundancy in the selected concepts. We propose the following modifications to the selection process that improve performance:

(Multi-Level Masking) We set all scores for a single image except the top-K to zero. Here, we evaluate three different levels, $K = 10$, $K = 50$, and $K = 100$, and we select the top-10 concepts for each level per side (A and B) and reduce them to 10 in a round-robin fashion (see Appendix B.4 for an ablation on the values for K). Intuitively, this allows the method to select concepts at different saliency levels. See Appendix E.2 for details.

(Redundancy-Aware Selection) To further eliminate redundancy in naive top-k selection, we select a diverse subset of concepts using Orthogonal Matching Pursuit (Residual-OMP). This greedily chooses concepts that best capture the unexplained variance in the binary set membership labels (in A or in B), i.e., at each iteration, Residual-OMP selects the concept that is most correlated with

	LVIS			MS COCO			SUN RGB-D			DOTA		
	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J
RAM++	31.5	56.9	35.9	44.7	62.8	44.1	15.1	<u>60.5</u>	39.7	29.4	53.6	31.4
MLLM + SigLIP (PTR)	26.4	41.9	27.1	40.2	48.5	37.3	10.1	42.6	23.5	33.8	41.7	30.2
MLLM + FG-CLIP (PTR)	22.2	39.8	23.9	27.4	40.6	25.9	15.8	42.1	27.4	31.1	40.8	27.0
+ Multi-Level Masking	27.5	43.1	27.5	42.4	49.4	38.2	20.6	47.7	37.3	32.4	40.0	27.6
+ Residual-OMP	32.8	45.8	33.2	53.5	54.2	48.5	22.9	49.9	44.2	45.5	47.0	41.1
+ MLLM Optimization	<u>39.6</u>	<u>61.0</u>	<u>42.9</u>	**68.0**	**75.7**	**66.4**	<u>24.3</u>	58.0	<u>46.9</u>	46.3	<u>55.2</u>	<u>43.7</u>
COCA + SigLIP	24.3	43.9	20.5	40.7	52.1	32.2	6.6	46.8	18.2	34.9	45.7	29.5
COCA + FG-CLIP	16.5	40.6	15.1	17.8	39.5	13.1	11.3	45.5	18.8	28.2	37.9	15.9
+ Multi-Level Masking	30.0	52.9	30.9	46.8	59.2	42.3	21.4	58.8	37.6	35.5	47.6	31.2
+ Residual-OMP	33.4	54.6	33.1	52.7	61.1	46.1	22.6	59.3	40.0	<u>47.7</u>	51.1	42.3
+ MLLM Optimization	**40.4**	**62.1**	**43.1**	<u>66.7</u>	<u>74.3</u>	<u>63.1</u>	**25.7**	**62.2**	**49.7**	**49.7**	**63.0**	**52.1**
Ground-Truth Concepts	59.4	–	–	96.3	–	–	36.8	–	–	58.4	–	–
Best 10 from COCA	65.8	–	–	96.3	–	–	41.7	–	–	76.3	–	–

Table 2: **Evaluation on noisy and multi-target datasets.** We compare concept discovery using different proposal sources (MLLM vs. COCA wordlist) and embedding models (SigLIP vs. FG-CLIP), and we progressively add our proposed Multi-Level Masking, Residual-OMP, and MLLM Optimization. **Bold** and <u>underlined</u> denote the best and second-best results, respectively.

the current residual, and then updates this residual by orthogonalizing the target labels against the subspace spanned by all previously selected concepts. See Appendix E.3 for details.

(Optimization) We use iterative MLLM optimization to find better concepts; see Section 3. Furthermore, here, we combine optimization with Residual-OMP. We prioritize MLLM-proposed concepts by adding a constant to their correlation score in Residual-OMP selection. This ensures that MLLM proposals are more likely to be included in the returned concepts.

***Results*.** We evaluate on MS COCO (Lin et al., 2014), LVIS (Gupta et al., 2019), SUN RGB-D (Song et al., 2015), and DOTA (Xia et al., 2018). For each dataset, we vary the number of images per set (150, 300, 500, and 1000) and the number of aligned classes per set (1 to 7), i.e., all 49 combinations of aligned classes.

Results are in Table 2. We observe that using fine-grained (FG-CLIP) embeddings requires multi-level masking to be effective. Without masking, FG-CLIP consistently underperforms SigLIP (e.g., 17.8% vs. 40.7% MCC-R for COCA-20k on MS COCO); however, when masking is applied, it outperforms SigLIP considerably. Also, we find that our proposed concept selection and optimization yield great improvements on all datasets over simple top-k selection. Residual-OMP improves MCC-R by up to $+11.1\%$ (MS COCO, MLLM). MLLM optimization further improves MCC-R on MS COCO to 68.0% and R_J to 66.4%. Finally, regarding the source of proposals, the static COCA wordlist is competitive against the more expensive MLLM-generated concept pool. Under the optimal configuration (Masking + Residual-OMP + Opt.), COCA outperforms MLLM proposals on LVIS, SUN RGB-D, and DOTA (e.g., 49.7% vs. 46.3% MCC-R on DOTA), while MLLM generation maintains a slight advantage only on MS COCO. Overall, these results show that finding distinguishing concepts in our complex and noisy datasets is very challenging, but our proposed methods substantially improve upon baselines. However, there remains a substantial gap to ground-truth concepts, which implies that this problem, unlike benchmarks in Section 4, is not yet solved.

6 Applications: Finding Spurious Correlations in ImageNet and Demographic Biases in T2I Models

***Finding Spurious Correlations in ImageNet*.** Prior work has shown that ImageNet contains class-associated visual features and spurious correlations that can be exploited by classifiers (Singla & Feizi, 2022; Neuhaus et al., 2023; Bykov et al., 2023). We use fine-grained image representations and image dataset diffing to systematically detect such class-concept correlations in ImageNet. Our approach involves multiple stages: First, we select concepts that activate more strongly for a class than for other classes, while filtering concepts that are semantically very similar to the class name. Then, we use an LLM to filter class-concept pairs based on interestingness and plausibility, removing low-quality or obvious correlations, such as ram (class) and pasture (concept). The LLM assigns

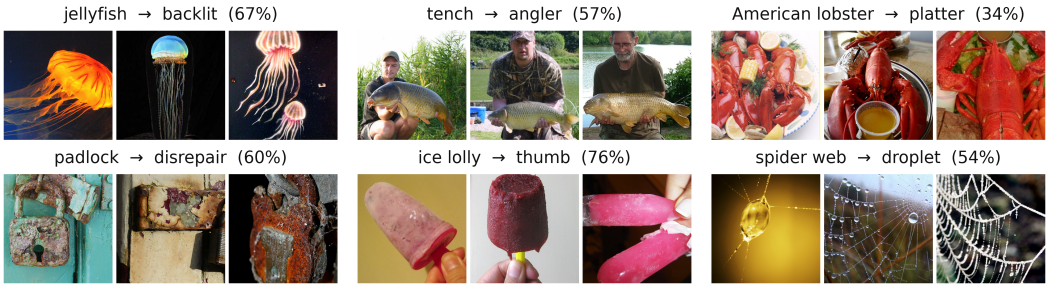


Figure 3: **Examples of spurious class-concept correlations in ImageNet.** For each class → concept pair, we show the ratio of class images containing the concept (according to Qwen3.8-27B) with three representative samples.

scores between 1 and 10 for how plausible and interesting a class-concept correlation would be. Finally, we verify the remaining candidates with an MLLM.

To filter concepts that activate more for a class than for other classes, we use dataset diffing in two stages. The first calculates the one-vs-rest AUROC, i.e., how well the concept separates the class from all other classes based on the FG-CLIP image-concept similarity scores. The second filter calculates the AUROC against the 20 classes with the most similar mean f_{emb} score vectors by cosine similarity. The second filter is necessary to eliminate associations such as dog-collar. "Collar" is discriminative for many dog classes relative to the entire dataset, but not discriminative for any dog class individually. Overall, for this application, we manually tuned the workflow to yield a manageable set of potentially interesting or unexpected correlations in ImageNet, not all correlations that a classifier could exploit. However, it can be adapted to new datasets or a higher number of candidates. See Appendix F.1 for more details.

Our results are as follows: We tested 6,103 candidate class-concept pairs in MLLM verification. Of these, 1,553 (25.4%) candidates across 578 classes survived, i.e., the concept is present in at least 10% of the class' images. We find a number of trends that involve multiple classes: fish presented as catch, i.e., held by a person (e.g., "tench" → "angler"); animals in zoo/captivity settings (e.g., "toucan" → "cage"); broken/decaying objects (e.g., "padlock" → "decrepit", "gas pump" → "rundown"); sea animals as seafood (e.g., "American lobster" → "boiled"); sports equipment with people or body parts (e.g., "rugby ball" → elbow). Furthermore, we find many individually interesting correlations such as sliced figs, graffiti on freight trains, or a surprisingly high number of flags with stars in the "flagpole" class. We show some examples in Fig. 3 and more in Appendix C.1.

Finally, we also run the entire workflow with SigLIP2 f_{emb} scores instead. SigLIP2 proposed 9,853 candidates, of which the MLLM verified 1,455 (14.8%, lower than the 25.4% acceptance rate for FG-CLIP). 609 concepts are proposed and verified both using SigLIP2 and FG-CLIP, but only 43.4% of FG-CLIP candidates (before verification) are included in SigLIP candidates. As expected, the omitted candidates concentrate on smaller attributes and objects, and surface textures. These results show that FG-CLIP is more precise in detecting correlations, and the represented themes are reasonably complementary between the models.

Finding Demographic Biases in Text-to-Image Models. We use FG-CLIP patch-level embeddings and image dataset diffing to find demographic biases in text-to-image (T2I) generative models, similar to (D'Incà et al., 2024). For this, we specifically target country, religion, and race categories and analyze which stereotypes text-to-image models associate with these categories, and how strongly they express them. As a basis, we select 10 countries (Egypt, Nigeria, Mexico, Brazil, United States, China, India, Turkey, Germany, Australia), 5 religions (Christianity, Judaism, Islam, Buddhism, Hinduism), and 7 race categories (Native American, Asian, Black, Hispanic, Middle Eastern, Pacific Islander, White). We select 60 occupations and 45 common activities and use two T2I models, Flux 1-dev (Flux) and Stable Diffusion 3.5 Large (SD-3.5), to generate 100 images for every prompt in the format "`a {country|religion|race adjective} {man|woman} is {working as occupation|doing activity }`". The country/religion/race specifier can be empty to create contrast sets.

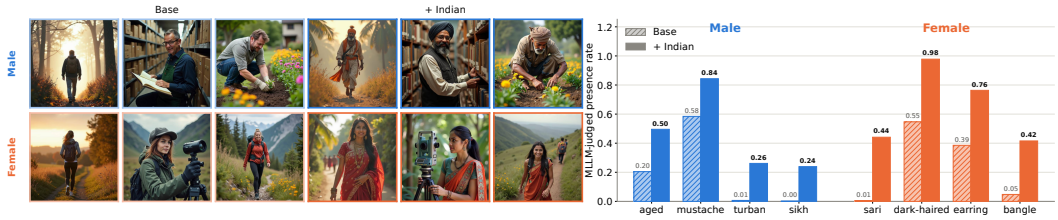


Figure 4: **Demographic bias detection example ("Indian").** *Left:* Generated images comparing base prompts against prompts with "+ Indian" across male (top) and female (bottom) subjects. *Right:* MLLM-judged presence rates for top discovered concepts in base (hatched) versus "+ Indian".

We then calculate the concept scores $f_{\text{emb}}(s, x)$ using FG-CLIP and SigLIP for the 5,317 visually concrete concepts, and contrast the image sets generated from prompts with country/religion/race specified, and prompts without their specification. We select 30 concepts per contrast set with Multi-Level Masking + Residual-OMP on FG-CLIP f_{emb} scores, the same method we use in Section 5. Then, we verify concept presence using an MLLM on 1800 (2×900) images per prompt pair, sampled uniformly across occupation/activity specifiers. See Appendix F.2 for further details on how we select and verify concepts.

We notice two general observations: (1) FG-CLIP and SigLIP largely disagree regarding the predicted bias concepts before MLLM verification, as their mean Jaccard index is only 2.4%. (2) The number of concepts that survive MLLM filtering is moderate for FG-CLIP (25.6%) and low for SigLIP (10.0%). This is also due to the strict difference threshold (10% between sets), which suppresses concepts with different distributions but low overall prevalence. Here, FG-CLIP still predicts more meaningful candidates; i.e., for 85% of FG-CLIP candidates, the MLLM attribution directions (whether an attribute is more frequent in images with country/religion/race specified or in images from the base distribution) agree with the FG-CLIP direction, but they agree for SigLIP candidates only 53% of the time. This makes SigLIP bias candidates effectively guesses.

We also compare Flux and SD-3.5, where in total we find 407 biased concepts for Flux and 541 for SD-3.5, i.e., SD-3.5 expresses more biases than Flux, while their strengths are comparable: the mean MLLM detection rate difference between prompts is 29.6% for SD-3.5 and 29.5% for Flux. Confirmed concepts largely differ (mean Jaccard index 25.4%). An example for "Indian" (country) is in Fig. 4. There, we find that Indian men are portrayed as older, more bearded, and wearing turbans, while Indian women are portrayed as black-haired and wearing Saris and jewelry. In Appendix C.2, we present more examples.

7 Conclusion

Our overarching goal is to understand differences between data distributions, specifically image distributions. In prior work, we found that methods typically rely on MLLM-proposed concepts, which we find costly, and on image-level embeddings, which are known for not representing details well. Intuitively, starting with a broad set of concepts that is cheap to evaluate and first fitting it to the current dataset should be more effective than proposing a potentially large number of concepts from small image samples. Therefore, in this paper, we evaluated and confirmed that MLLM proposals can be replaced by a static wordlist, and in many cases, patch embeddings have clear advantages over image-level embeddings. Furthermore, we found that standard benchmarks are simplistic and do not reflect real-world distributions. Therefore, we designed a setting that is more challenging, i.e., distributions are distinguished by multiple concepts with individually lower prevalence, and discovered that redundancy-aware concept selection, which has not been considered in prior work, is critical in this setting. Finally, we are interested in MLLM-driven optimization for understanding distributional differences and demonstrated that it can improve concept recall performance considerably. By communicating these findings, we significantly advance the field and the understanding of effective methods to interpret distributional differences. Beyond the methodological contributions, we also show how our insights enable useful applications with high practical value, for example, detecting spurious class-concept correlations or comparing outputs of text-to-image models.

AI USE STATEMENT

In this work, we used generative AI tools for generating synthetic data sets, helping develop theoretical models or conceptual frameworks, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, cleaning and reformatting datasets, supporting qualitative and thematic data analysis, and interpreting results. We have not used generative AI tools for formulating mathematical claims, assisting with translation, or assisting in the writing of proofs is not applicable to this work. Additionally, we used generative AI tools for suggesting experimental parameters, creating or editing software code, drafting parts of the paper, brainstorming, searching for information, and identifying relevant literature. We have reviewed all AI-assisted work. All cited literature has been read by the authors, and all AI-drafted text and code have been checked and revised by the authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This paper is concerned with improving methods that detect distributional differences between image datasets. We do not anticipate any particular concerns that arise from this research. Like all statistical models, embedding models and MLLMs can make mistakes, so outputs should be checked and verified by humans. We ablated all particular model choices for evaluation metrics with alternative models to mitigate artifacts from specific models.

Our experiment on demographic biases involves sensitive attributes such as race or religion. We acknowledge that AI models, including the multimodal embedding models and MLLMs that we use for bias detection, learn skewed representations and associations that may surface in downstream tasks, including bias detection. Therefore, in any deployment and downstream application, outputs must be checked. We also emphasize that attribute labels only concern *perceived* properties, because attributes like gender, race, or religion must be self-identified. Harms from this are mitigated by only labelling synthetic data, not images of real persons. Finally, we acknowledge that some demographic labels, especially race, are social constructs that have no equivalent in nature. We nonetheless include them in our analysis, as even social constructs can cause real-world harm.

REPRODUCIBILITY STATEMENT

This work uses publicly available data for all evaluations (VisDiffBench, MetaShift, LVIS, MS COCO, SUN RGB-D, DOTA, CelebA, ImageNet, BDD100K, Cityscapes). Additional artifacts and all code necessary to reproduce our results will be released upon acceptance of this paper. Prompts for LLM and MLLM models used in this work are included in the supplementary material alongside this submission.

REFERENCES

Michal Bravansky, Vaclav Kubon, Suhas Hariharan, and Robert Kirk. Dataset featurization: Uncovering natural language features through unsupervised data reconstruction. In *arXiv*, 2025.

Austin J. Brockmeier, Claudio Cesar Claros-Olivares, Matthew Emigh, and Luis Gonzalo Sanchez Giraldo. Identifying the instances associated with distribution shifts using the max-sliced bures divergence. In *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021.

Kirill Bykov, Laura Kopf, and Marina M-C Höhne. Finding spurious correlations with function-semantic contrast analysis. In *World Conference on Explainable Artificial Intelligence*, 2023.

Jinho Choi, Hyesu Lim, Steffen Schneider, and Jaegul Choo. Conceptscope: Characterizing dataset bias via disentangled visual concepts. In *NeurIPS*, 2025.

Mark Davies. The corpus of contemporary american english as the first reliable monitor corpus of english. In *Literary and Linguistic Computing*, 2010.

Boyang Deng, Songyou Peng, Kyle Genova, Gordon Wetzstein, Noah Snavely, Leonidas Guibas, and Thomas Funkhouser. Visual chronicles: Using multimodal llms to analyze massive collections of images. In *ICCV*, 2025.

Moreno D'Incà, Elia Peruzzo, Massimiliano Mancini, Dejia Xu, Vidit Goel, Xingqian Xu, Zhangyang Wang, Humphrey Shi, and Nicu Sebe. Openbias: Open-set bias detection in text-to-image generative models. In *CVPR*, 2024.

Lisa Dunlap, Yuhui Zhang, Xiaohan Wang, Ruiqi Zhong, Trevor Darrell, Jacob Steinhardt, Joseph E Gonzalez, and Serena Yeung-Levy. Describing differences in image sets with natural language. In *CVPR*, 2024.

Lisa Dunlap, Joseph E Gonzalez, Trevor Darrell, Fabian Caba Heilbron, Josef Sivic, and Bryan Russell. Discovering divergent representations between text-to-image models. In *ICCV*, 2025.

Sabri Eyuboglu, Maya Varma, Khaled Kamal Saab, Jean-Benoit Delbrouck, Christopher Lee-Messer, Jared Dunnmon, James Zou, and Christopher Re. Domino: Discovering systematic errors with cross-modal embeddings. In *ICLR*, 2022.

Chrisantha Fernando, Dylan Sunil Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. Promptbreeder: Self-referential self-improvement via prompt evolution. In *ICML*, 2024.

Arduin Findeis, Timo Kaufmann, Eyke Hüllermeier, Samuel Albanie, and Robert D Mullins. Inverse constitutional ai: Compressing preferences into principles. In *ICLR*, 2024.

Shiry Ginosar, Kate Rakelly, Sarah Sachs, Brian Yin, and Alexei A Efros. A century of portraits: A visual historical record of american high school yearbooks. In *ICCV Workshop*, 2015.

Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019.

Xinyu Huang, Youcai Zhang, Jinyu Ma, Weiwei Tian, Rui Feng, Yuejie Zhang, Yaqian Li, Yandong Guo, and Lei Zhang. Tag2text: Guiding vision-language model via image tagging. In *ICLR*, 2024.

Neha Hulkund, Nicolo Fusi, Jennifer Wortman Vaughan, and David Alvarez-Melis. Interpretable distribution shift detection using optimal transport. In *arXiv*, 2022.

Nicholas Jiang, Xiaoqing Sun, Lisa Dunlap, Lewis Smith, and Neel Nanda. Interpretable embeddings with sparse autoencoders: A data analysis toolkit. In *Mechanistic Interpretability Workshop at NeurIPS 2025*, 2025.

Younghyun Kim, Sangwoo Mo, Minkyu Kim, Kyungmin Lee, Jaeho Lee, and Jinwoo Shin. Discovering and mitigating visual biases through keyword explanation. In *CVPR*, 2024.

Yael Konforti, Mateo Espinosa Zarlenga, Elaf Almahmoud, and Mateja Jamnik. Cb-slice: Concept-based interpretable error slice discovery. In *arXiv*, 2026.

Sean Kulinski and David I Inouye. Towards explaining image-based distribution shifts. In *CVPR Workshop*, 2022.

Sean Kulinski and David I Inouye. Towards explaining distribution shifts. In *ICML*, 2023.

Weixin Liang and James Zou. Metashift: A dataset of datasets for evaluating contextual distribution shifts and training conflicts. In *ICLR*, 2022.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

Mingxuan Liu, Zhun Zhong, Jun Li, Gianni Franchi, Subhankar Roy, and Elisa Ricci. Organizing unstructured image collections using natural language. In *CVPR (Findings)*, 2026.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

Philip Naumann, Jacob Kauffmann, and Grégoire Montavon. Wasserstein distances made explainable: Insights into dataset shifts and transport phenomena. In *TPAMI*, 2026.

Yannic Neuhaus, Maximilian Augustin, Valentyn Boreiko, and Matthias Hein. Spurious features everywhere-large-scale detection of harmful spurious features in imagenet. In *ICCV*, 2023.

Tuomas Oikarinen, Subhro Das, Lam M. Nguyen, and Tsui-Wei Weng. Label-free concept bottleneck models. In *ICLR*, 2023.

Bruno A Olshausen and Michael R DeWeese. The statistics of style. In *Nature*, 2010.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *EMNLP-IJCNLP*, 2019.

Sahil Singla and Soheil Feizi. Salient imagenet: How to discover spurious features in deep learning? In *ICLR*, 2022.

Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *CVPR*, 2015.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. In *arXiv*, 2025.

Henrique Schechter Vera, Sahil Dua, Biao Zhang, Daniel Salz, Ryan Mullins, Sindhu Raghuram Panyam, Sara Smoot, Iftekhar Naim, Joe Zou, Feiyang Chen, et al. Embeddinggemma: Powerful and lightweight text representations. In *arXiv*, 2025.

Zihan Wang, Jingbo Shang, and Ruiqi Zhong. Goal-driven explainable clustering via language descriptions. In *EMNLP*, 2023.

Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Dota: A large-scale dataset for object detection in aerial images. In *CVPR*, 2018.

Chunyu Xie, Bin Wang, Fanjing Kong, Jincheng Li, Dawei Liang, Ji Ao, Dawei Leng, and Yuhui Yin. Fg-clip 2: A bilingual fine-grained vision-language alignment model. In *ICML*, 2026.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *ICLR*, 2024.

Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, 2020.

Youcai Zhang, Xinyu Huang, Jinyu Ma, Zhaoyang Li, Zhaochuan Luo, Yanchun Xie, Yuzhuo Qin, Tong Luo, Yaqian Li, Shilong Liu, et al. Recognize anything: A strong image tagging model. In *CVPR*, 2024.

Ruiqi Zhong, Heng Wang, Dan Klein, and Jacob Steinhardt. Explaining datasets in words: Statistical models with natural language parameters. In *NeurIPS*, 2024.

Hongyu Zhu, Sichu Liang, Wenwen Wang, Boheng Li, Tongxin Yuan, Fangqi Li, Hanyi Wang, Shi-Lin Wang, and Zhuosheng Zhang. Revisiting data auditing in large vision-language models. In *ACM Multimedia*, 2025.

Zhiying Zhu, Weixin Liang, and James Zou. Gsclip: A framework for explaining distribution shifts in natural language. In *arXiv*, 2022.

Supplementary Material

A VisDiffBench and MetaShift: Additional Ablations

A.1 Detailed Description of Metrics and Results with Qwen3.8-27B as Judge

We evaluate methods and baselines in Section 4 (Table 1) using three metrics:

Discrimination (D). To evaluate how informative the predicted set of concepts is regarding the difference between the sets A and B, we use an MLLM to classify all images (A and B combined) into the sets based on the 2×10 predicted concepts. For each image, we provide the MLLM with the image, the predicted concepts, and ask it to decide, according to the concepts, whether the image is in set A or in set B. After obtaining all per-image predictions, we calculate accuracy as the ratio of correct decisions.

Relevance ($Rel.$). To evaluate whether the predicted concepts actually apply to the images in the data, we use an MLLM to verify the presence of the 2×10 predicted concepts across the dataset. For each image, we provide the MLLM with the image and the list of predicted concepts, and ask it to output a boolean value for each concept indicating whether it is present in the image. After obtaining all per-image concept assignments, we calculate relevance as the average number of concepts the MLLM attributes to the images.

Embedding Similarity (Sim.). We calculate the maximum cosine embedding similarity between the ground truth concept and all predicted concepts for each side A and B, respectively. Then, we average the two per-side similarities. As an embedding model, we use the all-MiniLM-L6-v2 model from (Reimers & Gurevych, 2019).

In Section 4 (Table 1), we evaluate Discrimination and Relevance using Gemma4-31B as MLLM. Here, we additionally report results for Qwen3.8-27B (in FP8 quantization). They are in Table 3 and generally confirm our findings, i.e., they are not contingent on the judge model. Overall, we find that Qwen is slightly stricter in assigning attributes to images, so the relevance metric is shifted down slightly (mean $\Delta = -0.037$, max. $\Delta = 0.09$). Discrimination metrics stay mostly unchanged (mean $\Delta = 0.018$, max. $\Delta = 0.062$). For VisDiffBench, method rankings are preserved. One interesting finding is that on VDB-Hard, discrimination scores for Propose-then-rank, Embedding Sim. + Multimodal Optim., and True Concept are closer to each other, making their difference less notable. On the other hand, Embedding Sim. (without optimization) appears comparatively weaker.

For MetaShift, we see changes in ranking, but they appear among the already strong methods, i.e., Propose-then-rank, FG-CLIP, and Attention-Probing. Again, Attention-Probing without optimization appears comparatively weaker than under the Gemma4 evaluation.

	VDB-Easy			VDB-Med.			VDB-Hard			MetaShift		
	D	Rel	Rec	D	Rel	Rec	D	Rel	Rec	D	Rel	Rec
MLLM + SigLIP (PTR)	0.978	0.756	**0.549**	**0.880**	0.807	0.425	0.806	0.845	0.331	0.863	0.561	0.441
Distrib. Shift: U-WaX	0.963	0.313	0.306	0.799	0.258	0.185	0.714	0.214	0.147	0.805	0.229	0.400
Distrib. Shift: Linear	0.968	0.317	0.319	0.832	0.297	0.200	0.716	0.244	0.157	0.798	0.233	0.405
Caption Summary	0.968	0.584	0.338	0.843	0.713	0.267	0.794	0.640	0.236	0.822	0.403	0.252
Image Tagging: MLLM	0.968	0.552	0.346	0.858	0.635	0.288	0.783	0.608	0.243	0.817	0.392	0.310
Image Tagging: RAM++	0.972	**0.867**	0.335	0.816	**0.868**	0.262	0.715	**0.931**	0.219	0.838	**0.722**	0.424
COCA + FG-CLIP	0.959	0.522	0.283	0.832	0.502	0.203	0.724	0.456	0.134	**0.878**	0.472	**0.501**
COCA + SigLIP	0.947	0.632	0.336	0.809	0.639	0.275	0.750	0.645	0.275	0.723	0.403	0.306
+ MLLM Optim.	0.981	0.690	0.511	0.868	0.681	**0.445**	**0.808**	0.769	**0.402**	0.757	0.397	0.307
+ FG-CLIP	**0.982**	0.668	0.489	0.877	0.704	0.417	0.805	0.681	0.341	0.857	0.490	0.465
True Concept	0.980	-	-	0.880	-	-	0.818	-	-	0.891	-	-

Table 3: Replication of Table 1 using Qwen3.8-27B as MLLM judge for D and Rel metrics.

A.2 Ablating the MLLM for SigLIP Embedding Similarity with Optimization

In Table 1, we used Gemma4-31B as MLLM and disabled reasoning for multimodal optimization. While this performs well on VisDiffBench, results on MetaShift are worse than other methods. We attribute this to using image-level SigLIP embeddings, which do not represent the fine-grained details in MetaShift sufficiently well. However, Propose-then-Rank also uses SigLIP embeddings, but performs well also on MetaShift. Therefore, the reason is not only the embedding space, but rather how it interacts with the MLLM. In order to show that this is not an artifact of using Gemma4-31B, we ablate the following variations on MetaShift (Embedding Sim. + Multimodal Optim.): (1) Gemma4-31B with reasoning enabled; (2) Gemma4-31B with high-resolution image input; (3) Qwen3.6-27B with reasoning enabled; (4) MiniMax M3 and MiMo-V2.5 as frontier LLMs.

Results for discrimination (D, using Gemma4-31B as judge) are in Fig. 5. We see that every setting improves over the baseline value reported in Table 1 (0.757). However, performance is still worse generally than any other method, even for frontier MLLMs. This shows that the feature initialization with SigLIP embeddings is problematic in a way that any MLLM struggles to recover useful information about the dataset difference.

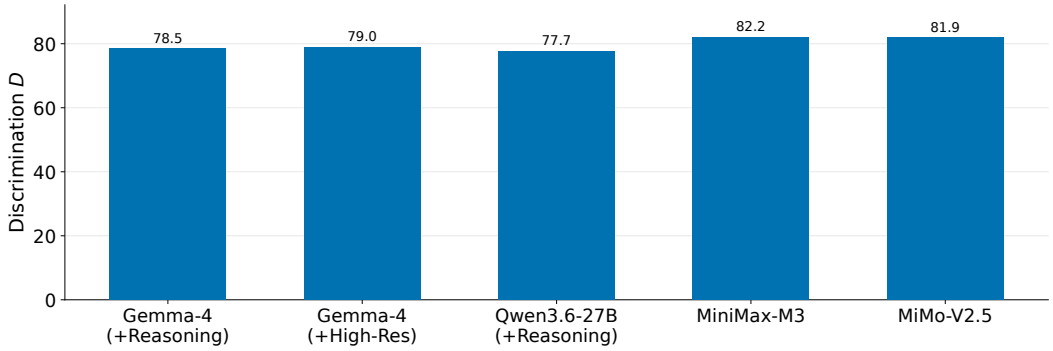


Figure 5: MLLM ablation on MetaShift. Discrimination performance (D) of the Embedding Sim. + Multimodal Optim. method across different MLLMs and inference configurations.

A.3 Evaluating Embeddings Models on Fine-Grained Perception

We analyse the concrete advantage of FG-CLIP in detecting fine-grained attributes. Given a dataset with instance-level annotations, we convert the annotations into image-level presence labels, i.e., whether an attribute or object is present in the image, so the task is a per-attribute binary classification task. We use six datasets that provide such presence labels derived from their own annotations: BDD100K (Yu et al., 2020) (driving scenes; small road objects and lane-marking types), DOTA (Xia et al., 2018) (aerial imagery; very small objects in large tiles), CelebA (Liu et al., 2015) (facial attributes, i.e., non-object properties), SUN RGB-D (Song et al., 2015) (cluttered indoor scenes), MS COCO (Lin et al., 2014) and LVIS (Gupta et al., 2019) (common objects). From each dataset, we draw at most 10,000 images and only classes whose prevalence in the sample lies in $[10\%, 90\%]$. The $10,000$ images are selected in a way that maximizes the number of classes that fulfill this criterion.

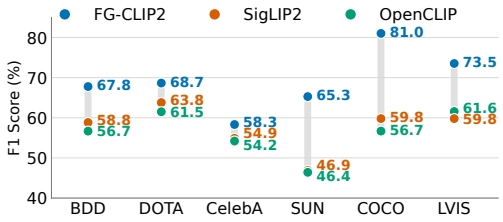


Figure 6: Fine-grained attribute detection performance (F1 score) across six datasets.

For each attribute class, we calculate the F1 score based on class name (as defined in the dataset), and based on the best concept among the 54K words used in Section 4 according to F1 score. For each model and dataset, we individually calculate the per-attribute threshold on the image-text embedding

similarities that maximises the F1 score. We compare three model families, i.e., FG-CLIP, SigLIP2, and OpenCLIP.

Results are in Fig. 6. On class-name-based classification, the best FG-CLIP2 checkpoint reaches an average F1 of 69.1, against 57.3 for the best SigLIP2 and 56.2 for the best OpenCLIP model. This demonstrates a clear advantage of FG-CLIP on fine-grained detection. The advantage is most pronounced on object detection datasets like COCO, LVIS, and SUN-RGBD, and less pronounced on CelebA, DOTA, and BDD100K, which include less complex scenes. This means that SigLIP and CLIP models have the ability to represent small objects, but they get increasingly suppressed when scenes get more complex. We conclude that there is a genuine representational difference between FG-CLIP patch embeddings and image-level embeddings, and this difference surfaces in realistic datasets, so we can expect it to appear in reality.

A.4 Ablating the Prevalence of Target Concepts for Propose-then-Rank

Table 1 shows that propose-then-rank (using SigLIP for ranking) performs well on both MetaShift and VisDiffBench. However, in theory, this method's performance scales negatively with the prevalence of target concepts in the data, because it does not use a global concept list for guidance, but derives all concepts from small samples (e.g., 2×5 images) of the data via MLLM proposals. In this case, the probability of seeing an image with the target concept is $1 - (1 - p)^k$, where p is the prevalence of the target concept, i.e., the number of images in the set that have the concept, and k is the number of images per sample. Additionally, even when the target concept is in a sample, there is no guarantee the MLLM will propose it as a candidate. Therefore, we analyze propose-then-rank for different prevalence strengths on the ImageNet and LVIS datasets. For all prevalence strengths from 2% to 50% in 2% steps, we sample 5 class pairs from each dataset and construct, for each pair, datasets with 50, 100, 200, 300 and 500 images per set (A and B). The sampled classes are the target concepts and become the majority class of their respective set, i.e., no other class is represented more frequently in that set. The remaining images are filled randomly from other classes, with the majority constraint. From each dataset, we sample 100 subsamples of 2×5 images (5 from each set) and let the MLLM propose 10 candidate concepts for each set. These are subsequently ranked with SigLIP2 and FG-CLIP, and we select the 10 highest scoring concepts. We then ask an LLM judge whether a set of candidates (both the per-sample and the final top-ranked candidates) contains the ground-truth class or a closely related expression. From this, we calculate two metrics: The expected hit rate, i.e., the likelihood of predicting the ground-truth class from an individual sample, and the final ranking quality, i.e., whether the ground-truth concept is in the 10 final returned concepts for a set.

Results are in Fig. 7. We find that indeed the hit rate is small for $p < 5\%$, but monotonically increases with prevalence. The final score (after 100 samples) increases sharply below $p < 10\%$ and only slowly increases or stagnates for higher prevalences. This explains the strong performance of propose-then-rank, as for even somewhat prevalent concepts ($\approx 10\%$ of images), the MLLM will find it with enough samples. However, the variance between concepts is high, as some are easier to detect than others, and our judge is rather strict on concept equivalence. Finally, we note that on LVIS, as expected, FG-CLIP yields better concept recall, while SigLIP and FG-CLIP perform similarly on ImageNet with a small advantage for SigLIP. This shows that propose-then-rank can also benefit from fine-grained perception.

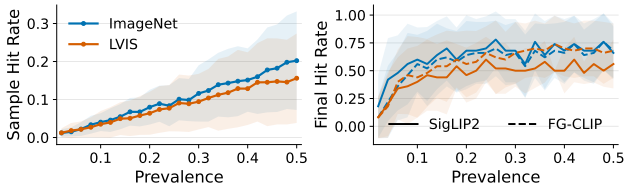


Figure 7: Propose-then-rank effectiveness across target concept prevalences. Shaded regions indicate variance.

A.5 Testing MLLMs on Fine-Grained Concept Detection

Throughout this paper, we use MLLMs (Gemma4-31B and Qwen3.8-27B) to evaluate methods, mostly through verifying that concepts are present or absent in given images. To show that Gemma4-

31B and Qwen3.8-27B yield reliable concept indicators, we report their classification performance on MS COCO, SUN RGB-D, CelebA, DOTA, and BDD100K. From each dataset, we select classes that appear in at least 10% of images and in at most 90% of images. For each class, we sample 500 positive images, i.e., images where the class is present, and 500 negative images, i.e., images where the class is absent, and use the MLLM to predict the class presence.

In Table 4, we report accuracy, precision, recall, and F1 score. We find that Gemma4-31B and Qwen3.8-27B perform almost identically on all datasets, reaching around 80% accuracy on average. On BDD100K, DOTA, CelebA, and SUN RGB-D, accuracy is around 80%. On MS COCO, it is higher, $\approx 90\%$ for Gemma4-31B and $\approx 88\%$ for Qwen3.8-27B. The similar performance across all datasets, including the better performance on MS COCO, indicates that remaining errors are dataset ambiguities or annotation problems instead of systematic MLLM errors.

	Gemma4-31B				Qwen3.8-27B			
Dataset	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
BDD100K	0.784	0.893	0.676	0.758	0.799	0.930	0.649	0.760
DOTA	0.778	0.930	0.619	0.721	0.788	0.948	0.610	0.730
CelebA	0.792	0.821	0.778	0.762	0.777	0.815	0.735	0.736
SUN RGB-D	0.810	0.818	0.828	0.813	0.808	0.839	0.785	0.802
MS COCO	0.904	0.968	0.836	0.890	0.879	0.988	0.768	0.855
Average	0.800	0.863	0.747	0.775	0.797	0.877	0.711	0.763

Table 4: MLLM binary detection accuracy, macro-averaged over classes inside each dataset. Evaluation uses up to 500 positive + 500 negative sampled images per class.

B Multi-Target Datasets: Additional Ablations

B.1 Ablating Residual-OMP for Redundancy-Aware Concept Selection

In Section 5, we propose to use Residual-OMP (see Appendix E.3) to select diverse concepts instead of simply selecting the top-K highly scoring concepts. However, it is not difficult to come up with many variations that fulfill similar purposes. Therefore, in Table 5, we compare top-k selection and Residual-OMP to a number of similar methods (which we briefly describe below), and find that, while many methods perform similarly well, Residual-OMP is the best method on average across datasets according to both settings (MLLM and COCA wordlist) and all metrics.

		LVIS			MS COCO			SUN RGB-D			DOTA			Avg.		
		MCC-R	Sim.	R_J	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J
MLLM	Top-K	27.5	43.1	27.5	42.4	49.4	38.2	20.6	47.7	37.3	32.4	40.0	27.6	30.7	45.1	32.7
	MMR	32.7	45.7	32.9	52.9	54.0	48.2	22.5	49.5	43.6	42.5	45.7	37.6	37.7	48.7	40.6
	mRMR	32.3	45.6	32.9	51.9	53.9	47.3	22.7	49.5	42.9	40.2	44.7	34.9	36.8	48.4	39.5
	mRMR cross-level	30.8	44.9	32.3	50.4	52.4	45.5	21.2	49.0	42.3	39.5	44.7	34.7	35.5	47.8	38.7
	Forward Selection	29.2	44.0	29.3	46.1	50.2	40.7	21.2	49.1	41.9	38.3	42.7	32.6	33.7	46.5	36.1
	Cluster-then-Select	31.0	45.0	31.7	51.6	53.3	46.2	21.4	49.4	43.4	44.1	46.0	39.2	37.1	48.4	40.1
	Facility location	24.3	39.2	17.5	35.7	42.4	24.7	20.3	47.4	34.5	42.8	44.9	39.4	30.8	43.5	29.0
	ℓ_1 LogReg	31.7	45.4	32.5	52.3	54.1	48.1	22.1	49.3	42.9	42.8	45.7	37.9	37.2	48.6	40.4
	Residual-OMP	**32.8**	**45.8**	**33.2**	**53.5**	**54.2**	**48.5**	**22.9**	**49.9**	**44.2**	**45.5**	**47.0**	**41.1**	**38.6**	**49.2**	**41.8**
COCA-20k	Top-K	30.0	52.9	30.9	46.8	59.2	42.3	21.4	58.8	37.6	35.5	47.6	31.2	33.4	54.6	35.5
	MMR	33.1	54.2	32.4	52.3	60.9	45.9	22.0	58.9	39.5	42.9	49.9	38.3	37.6	56.0	39.0
	mRMR	33.3	**54.8**	**33.5**	52.1	61.1	46.3	22.5	59.3	40.0	41.5	49.9	36.6	37.4	56.3	39.1
	mRMR cross-level	31.0	52.8	30.9	48.5	58.3	41.6	21.1	58.2	38.0	39.7	48.6	35.2	35.1	54.5	36.4
	Forward Selection	31.4	53.7	31.1	46.4	57.6	38.8	21.9	**60.3**	**40.6**	39.8	48.6	35.5	34.9	55.1	36.5
	Cluster-then-Select	31.7	53.3	31.1	50.0	59.6	43.8	21.1	58.3	37.4	43.5	48.6	37.2	36.6	55.0	37.4
	Facility Location	26.6	48.2	21.4	38.8	50.9	28.0	17.0	53.2	27.2	40.3	50.0	33.8	30.7	50.6	27.6
	ℓ_1 LogReg	33.0	54.6	33.3	52.5	**61.5**	**46.8**	**22.6**	59.4	39.8	44.7	50.7	40.6	38.2	**56.6**	40.1
	Residual-OMP	**33.4**	54.6	33.1	**52.7**	61.1	46.1	**22.6**	59.3	40.0	**47.7**	**51.1**	**42.3**	**39.1**	56.5	**40.4**

Table 5: Comparing Residual-OMP as redundancy-aware selection method to alternative methods.

The compared methods are defined as follows:

MMR. Maximal Marginal Relevance iteratively selects concepts by maximizing a trade-off between the candidate's score divergence $\delta_{A,B}(s)$ and its redundancy with already selected concepts. Redun-

dancy is measured as the maximum correlation between the standardized scores of a candidate and those of any previously selected concept.

mRMR. Minimum Redundancy Maximum Relevance operates identically to MMR, but computes the redundancy penalty differently. Instead of penalizing the maximum correlation, mRMR penalizes the mean correlation between the candidate and all previously selected concepts.

mRMR cross-level. This variant of mRMR is specifically designed to improve the multi-level masked concept selection described in Appendix E.2. Instead of selecting concepts independently for each masking level K, it carries over the selection history from earlier levels. Candidates chosen at earlier levels can never be re-picked, and the method explicitly penalizes similarity to them, which frees up selection budget for new concepts.

Forward Selection. Forward sequential selection iteratively selects the set of concepts to maximize joint predictive performance. At each step, it adds the remaining candidate that yields the highest cross-validated ROC-AUC when combined with the already selected concepts using a logistic regression classifier to predict set membership (A vs. B).

Cluster-then-Select. This method groups the candidate concepts into exactly k clusters based on their pairwise score distances (computed as 1 minus the correlation) using agglomerative clustering with average linkage. For each cluster, the concept with the highest score divergence $\delta_{A,B}(s)$ is chosen as the representative. This naturally yields k distinct concepts while retaining the most discriminative features within each cluster.

Facility Location. This approach models concept selection as a submodular maximization problem based on the facility location objective. It iteratively selects candidates that maximize the marginal gain in covering the similarity matrix of the candidate pool, weighted by the non-negative score divergence of the candidates.

ℓ_1 ***LogReg.*** This method fits an ℓ_1-regularized logistic regression model to predict the binary set membership labels using the standardized score matrix as input. We search over a grid of regularization strengths to find the value that yields exactly k non-zero coefficients. The corresponding concepts are then selected and ordered by the absolute magnitude of their learned weights.

B.2 Ablating the Number of Predicted Concepts per Side

In Section 5, we evaluate all methods based on 10 predicted concepts per side (A and B), i.e., 20 concepts in total. However, this is an arbitrary choice whose main utility lies in its compactness, which makes it easily understandable by human users, and 10 concepts per side exceeds the maximum number of per-side aligned concepts in the dataset construction, which is seven. To validate that our findings are not an artifact of this particular choice of the number of predicted concepts, we evaluate all methods using $k = 5, 10, 15, 20, 30$ concepts per side. Results are in Table 6 (averaged across datasets). Naturally, recall numbers correlate with the number of selected concepts, but the relative ordering of methods and their margins stay the same.

	$k=5$			$k=10$			$k=15$			$k=20$			$k=30$		
	R@k	Sim.	R_J	R@k	Sim.	R_J	R@k	Sim.	R_J	R@k	Sim.	R_J	R@k	Sim.	R_J
MLLM + FG-CLIP	20.3	36.1	20.9	24.1	40.8	26.0	26.7	43.7	30.6	28.7	45.9	34.0	31.7	48.9	39.3
+ Multi-Level Masking	25.6	39.7	25.4	30.7	45.1	32.7	34.1	48.4	38.5	36.6	50.8	42.7	40.6	54.3	49.8
+ Residual-OMP	30.5	42.1	30.0	38.6	49.2	41.8	42.4	52.8	49.2	44.6	54.9	53.6	47.6	57.9	60.3
+ MLLM Optimization	36.1	54.4	36.8	44.5	62.3	49.5	48.3	66.1	56.5	50.7	68.5	61.4	53.3	71.6	68.0
COCA + SigLIP	22.0	42.4	19.4	26.6	47.1	25.1	30.0	49.8	29.7	32.6	51.8	33.3	35.9	54.7	38.9
COCA + FG-CLIP	14.9	36.7	11.2	18.4	40.9	15.7	20.7	43.3	18.8	22.4	45.0	20.9	24.7	47.5	24.7
+ Multi-Level Masking	27.4	48.7	26.9	33.4	54.6	35.5	36.9	58.0	41.2	39.6	60.5	45.8	43.6	63.9	52.5
+ Residual-OMP	31.4	50.2	29.7	39.1	56.5	40.4	42.9	59.5	46.4	45.3	61.5	50.6	48.4	64.0	56.7
+ MLLM Optimization	**36.5**	**56.9**	**38.2**	**45.1**	**65.0**	**51.2**	**49.0**	**68.9**	**58.4**	**51.6**	**71.5**	**63.5**	**54.9**	**74.7**	**69.7**
Oracle (ground-truth concepts)	62.7	–	–	62.7	–	–	62.7	–	–	62.7	–	–	62.7	–	–
Oracle (best k from COCA)	70.0	–	–	70.0	–	–	70.0	–	–	70.0	–	–	70.0	–	–

Table 6: Evaluating methods from Section 5 with varying number of predicted concepts per side. Metrics averaged across datasets.

B.3 Ablating the Image Sampling in MLLM Optimization

In Section 5, we optimize concepts using an MLLM by randomly sampling images from sets A and B and asking the MLLM to propose new concepts that describe the difference between the sets. An alternative is to specifically select images that are not explained by the already selected concepts. Here, we ablate how useful this modification is against sampling random images for the MLLM. The results are in Table 7. We find that both sampling strategies perform similarly, but random sampling performs marginally better. We conclude that optimizing the sampling process for MLLM optimization is nontrivial.

	LVIS			MS COCO			SUN RGB-D			DOTA		
	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J
MLLM + Optimization	39.6	61.0	42.9	68.0	75.7	66.4	24.3	58.0	46.9	46.3	55.2	43.7
+ unexplained images	39.4	60.4	42.5	67.6	75.2	65.3	24.2	57.8	47.1	46.2	54.9	43.7
COCA + Optimization	40.4	62.1	43.1	66.7	74.3	63.1	25.7	62.2	49.7	49.7	63.0	52.1
+ unexplained images	39.8	61.6	42.5	65.7	74.0	62.0	26.0	62.2	49.7	48.4	63.3	51.7

Table 7: Comparison of MLLM-based concept optimization. In the standard setting, we randomly sample images. In the evaluated variant, we specifically select concepts that are not explained by the current concept selection.

B.4 Ablating the Number of Nonzero Concepts in Multi-Level Masking

In Section 5, we propose to select less redundant concepts by multi-level masking, i.e., keeping only the top-10, top-50, and top-100 concept scores per image and setting all others to zero. Here, we ablate whether this method is sensitive to the exact choice of top-10/50/100, and how many masking rounds there are. For both COCA and MLLM-proposed concept pools, we evaluate the multi-level masking strategy for concept selection with 20 rounds of randomly selected top-k triples, where we sample the first k (k_1) uniformly in $[5, 20]$, k_2 in $[30, 70]$, and k_3 in $[80, 120]$. Additionally, we evaluate 20 rounds where we include four masking levels, with top-k values sampled as k_1 in $[5, 15]$, k_2 in $[20, 40]$, k_3 in $[50, 70]$, and k_4 in $[80, 120]$.

Results are in Table 8. We find that evaluations with sampled top-k values, including four levels, deviate only minimally from the (10, 50, 100) values used in Table 2 and min/max ranges overall are tight. We conclude that the choice of levels and the top-k values is less important than masking overall.

	LVIS			MS COCO			SUN RGB-D			DOTA		
	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J	MCC-R	Sim.	R_J
MLLM + FG-CLIP + Multi-Level Masking												
Published $(10, 50, 100)$	27.5	43.1	27.5	42.4	49.4	38.2	20.6	47.7	37.3	32.4	40.0	27.6
3 levels, sampled	$27.4_{27.0\text{–}28.1}$	$43.1_{42.8\text{–}43.3}$	$27.7_{26.9\text{–}28.3}$	$42.1_{41.4\text{–}42.7}$	$49.3_{49.0\text{–}49.8}$	$38.2_{37.4\text{–}39.2}$	$20.6_{20.0\text{–}21.1}$	$47.6_{47.3\text{–}47.8}$	$36.7_{36.2\text{–}37.3}$	$32.5_{31.7\text{–}34.1}$	$40.3_{40.1\text{–}40.6}$	$28.0_{27.2\text{–}29.4}$
4 levels, sampled	$27.3_{26.8\text{–}28.0}$	$43.0_{42.8\text{–}43.2}$	$27.6_{27.0\text{–}27.8}$	$42.1_{41.4\text{–}42.5}$	$49.3_{49.0\text{–}49.5}$	$38.0_{37.4\text{–}38.5}$	$20.5_{19.8\text{–}21.2}$	$47.6_{47.4\text{–}47.9}$	$36.8_{36.2\text{–}37.5}$	$32.4_{31.7\text{–}33.8}$	$40.1_{39.8\text{–}40.3}$	$27.9_{27.0\text{–}29.7}$
COCA + FG-CLIP + Multi-Level Masking												
Published $(10, 50, 100)$	30.0	52.9	30.9	46.8	59.2	42.3	21.4	58.8	37.6	35.5	47.6	31.2
3 levels, sampled	$30.3_{29.7\text{–}30.9}$	$53.1_{52.5\text{–}53.5}$	$30.8_{29.3\text{–}32.0}$	$46.3_{45.7\text{–}46.9}$	$59.1_{58.6\text{–}59.4}$	$42.0_{40.6\text{–}42.8}$	$21.6_{21.0\text{–}22.2}$	$59.1_{58.8\text{–}59.3}$	$37.6_{36.8\text{–}38.0}$	$35.2_{34.6\text{–}36.0}$	$47.7_{47.5\text{–}47.9}$	$31.6_{30.8\text{–}32.4}$
4 levels, sampled	$30.4_{29.8\text{–}30.9}$	$53.1_{52.7\text{–}53.4}$	$30.9_{30.1\text{–}31.9}$	$46.0_{45.5\text{–}46.7}$	$58.9_{58.3\text{–}59.1}$	$41.7_{40.7\text{–}42.4}$	$21.7_{21.1\text{–}22.1}$	$59.1_{58.9\text{–}59.4}$	$37.7_{36.6\text{–}38.8}$	$35.2_{34.5\text{–}35.8}$	$47.6_{47.4\text{–}47.8}$	$31.5_{30.6\text{–}32.4}$

Table 8: **Sensitivity to the masking levels.** Multi-Level Masking with the published per-image top-K levels $(10, 50, 100)$ against 20 randomly sampled level sets per scheme. Sampled rows report the mean over the 20 configurations, with the range (min-max) as a subscript. All other settings (FG-CLIP, Top-K selection, 10 concepts per side) are unchanged.

C Applications: Additionaly Results

C.1 Additional ImageNet Spurious Correlations

In Section 6, we use FG-CLIP embeddings and dataset diffing to find candidates for spurious class-concept correlations in ImageNet. Of the 6,103 proposed candidates, we verified 1,553 class-concept pairs with an MLLM. This still contains a mix of genuinely interesting correlations, obvious or expected associations, and borderline and ambiguous cases. Therefore, the final list requires manual inspection. Furthermore, many correlations are specific to one class and one concept and cannot be

generalized, which makes a systematic presentation impossible. Instead, we selected eight of the discovered correlations that we found interesting in Fig. 8.

Specifically, we found a large number of freight trains with graffiti on them (65% MLLM presence score), which aligns with the "decay" theme of objects. We found that many doormats have words on them (60% presence score), some "maypole" images show traditional female dresses (58%), and many prayer rugs show a diamond or route pattern (51%), which has symbolic meaning in Muslim cultures. One very interesting example is the high number of stars on flags in the "flagpole" class. This is not exclusive to US "stars-and-stripes" flags, but also includes, for example, the Turkish, Californian, and Texan flags. Finally, we found many bullet train images with steel constructions (trusses), which is perhaps less surprising, but also many ovens with fire burning and sliced figs, which are peculiar distributions.

C.2 Additional Results for Demographic Bias Detection in T2I Models

In Fig. 9, we show additional examples of highly significant demographic biases in images generated by Flux and SD-3.5-Large. For countries, we show examples for "Chinese" (Fig. 9a) and "German" (Fig. 9b). We find that, relative to the base prompt, Chinese men are portrayed with dark hair and less facial hair, while German men are portrayed older and less with dark hair. For women, we find that the same haircolor and age pattern holds. Another interesting finding is specific accessories, such as "jacket" for German men and "hairpin" for Chinese women. The distributional differences may be induced by genuinely different demographics, cultural stereotypes, or skewed sampling of the training data. Assessing which biases are harmful or undesired is outside of the scope of this work.

Another interesting example is "Pacific Islander" as a race group (Fig. 9c). Both men and women are portrayed as showing more bare skin (instantiated by the reduced "long-sleeved" attribute), and Pacific Islander women are portrayed as wearing more earrings and necklaces. The increased prevalence of jewelry for women, together with the increase of "smiling" for both men and women, is a recurring pattern for demographic groups, especially nonwestern groups.

Finally, we compare outputs for religious groups. We find that all religions except "Christian" are portrayed in a very marked way. For example, "Buddhist" persons are almost exclusively displayed as monks or nuns. "Jewish" prompts generate mostly images where people look "Hasidic" (Fig. 9f), e.g., men wearing hats ("fedora") and women's bonnets. Both genders overall show less hair and are portrayed older. Across most demographic groups, age shifts are among the most significant shifts. Images for "Christian" (Fig. 9d) overall differ less from the base distribution, but include more dark-skinned people ("African-American", "ebony") and women appear older (20% less prevalence of "girl"). Finally, we noticed that "atheist" (Fig. 9e) images also differ from the base distribution in a peculiar way: Men appear older, more balding, and wearing more glasses. Women appear "expressionless" and directly face the viewer ("stare"). Both groups are portrayed less happy.

D Detailed Description of Methods

D.1 Propose-then-Rank

The propose-then-rank baseline follows Dunlap et al. (2024) and consists of two stages: (i) MLLM-driven candidate generation from visual prompts, and (ii) embedding-based ranking over the full image sets.

Candidate Generation ("Propose"). Let k denote the number of proposal iterations (we use $k = 5$). In each iteration $r \in \{1, \dots, k\}$, we randomly sample five images from set A and five images from set B. These images are arranged in a 2×5 grid, with set A in the top row and set B in the bottom row. The grid is presented to an MLLM together with a prompt requesting n short visual concepts that characterize set A relative to set B, and n concepts that characterize set B relative to set A (we use $n = 10$). The MLLM returns a structured list of candidate phrases, such as "outdoor scene" or "bright lighting". After k iterations, we collect all proposed candidates and deduplicate them to obtain a vocabulary $\mathcal{S}$ of unique concepts.

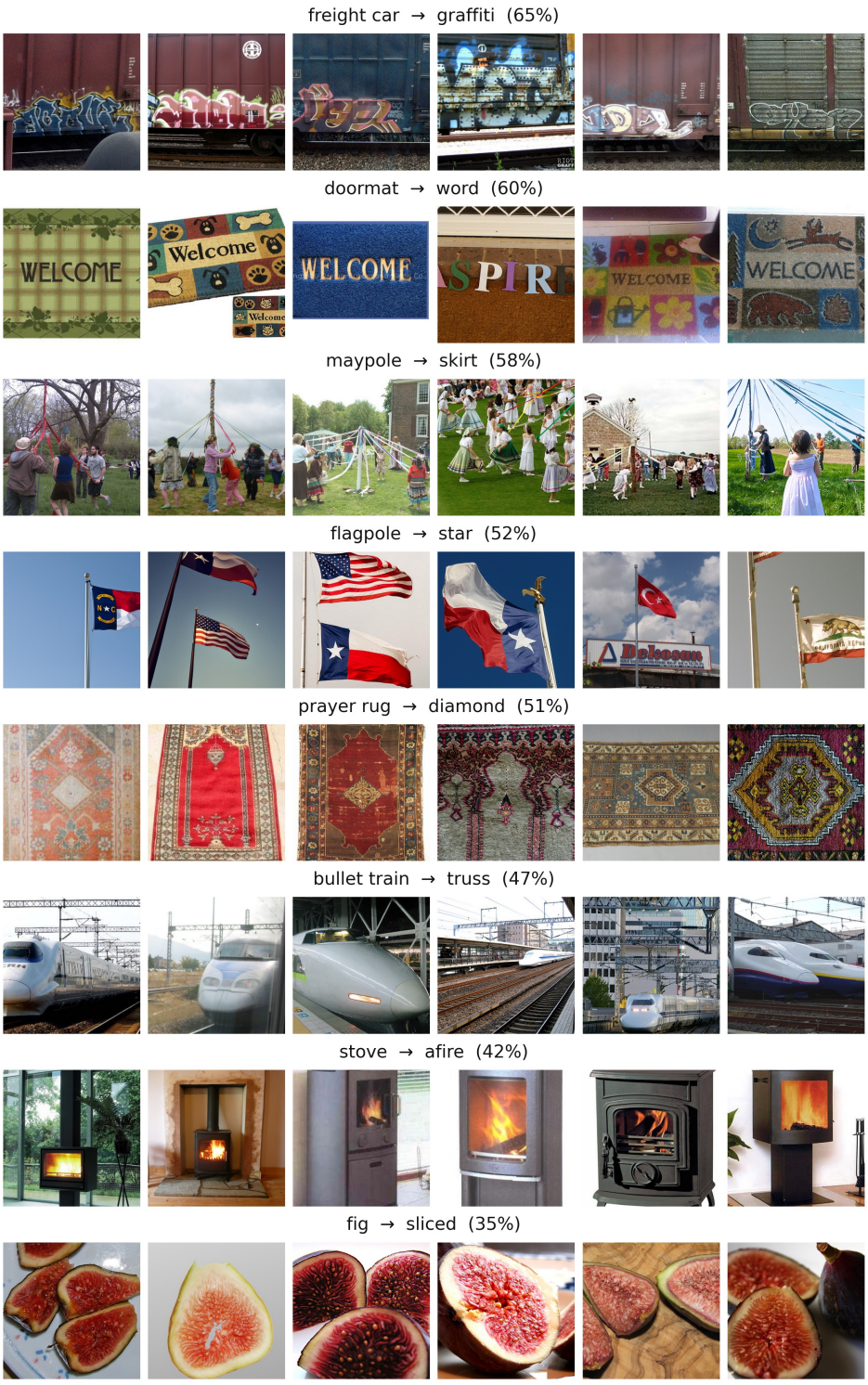
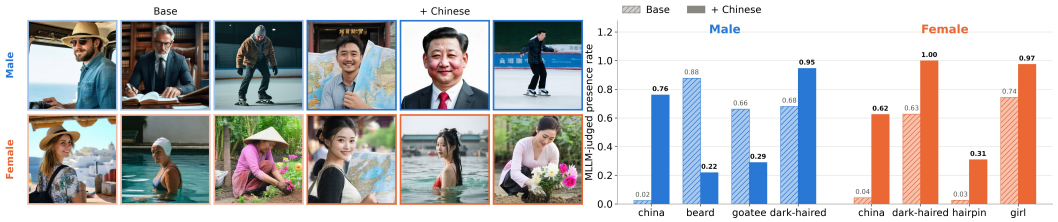
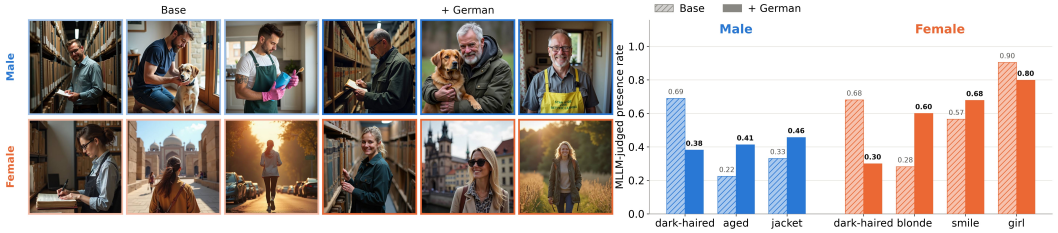


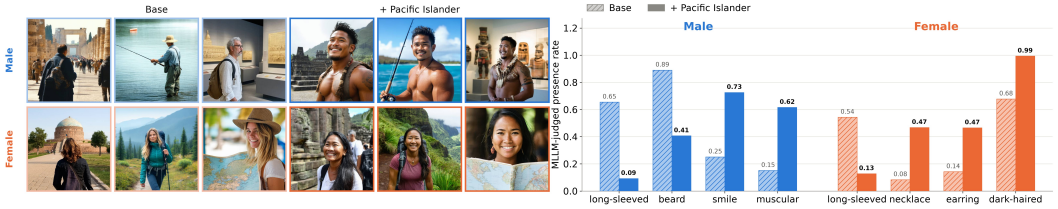
Figure 8: Examples of discovered class–concept correlations in ImageNet. Each row displays representative images for a class alongside the correlated concept and its MLLM-verified presence score (in parentheses).



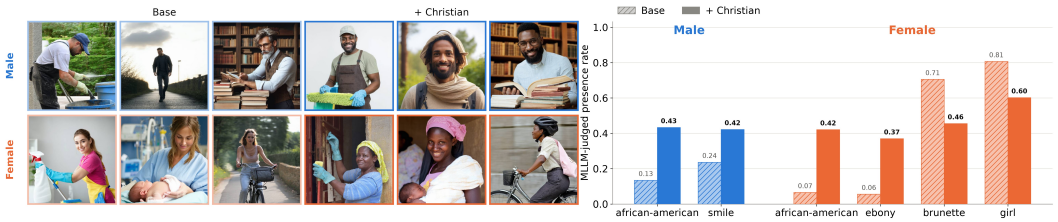
(a) Example detected biased attributes for "Chinese" (country) demographic group.



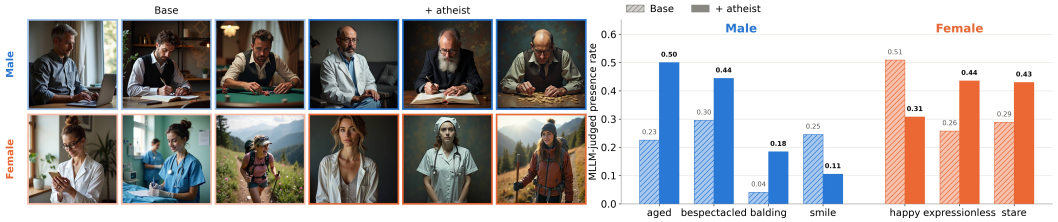
(b) Example detected biased attributes for "German" (country) demographic group.



(c) Example detected biased attributes for "Pacific Islander" (race) demographic group.



(d) Example detected biased attributes for "Christian" (religion) demographic group.



(e) Example detected biased attributes for "atheist" (religion) demographic group.



(f) Example detected biased attributes for "Jewish" (religion) demographic group.

Figure 9: Examples of detected biased attributes across country/religion/race demographic groups across Flux and SD-3.5-Large T2I models.

Ranking. For ranking, we evaluate every candidate $s \in \mathcal{S}$ against all images in both sets using image-text similarity. Let $\phi(x)$ denote the normalized image embedding of image x and $\psi(s)$ the normalized text embedding of concept s, both obtained from SigLIP2. We calculate $f_{\text{emb}}(s, x)$ (see Section 3).[1]

For each candidate s, we compute its directional discriminativeness via the area under the ROC curve (AUROC), following Dunlap et al. (2024). We find that when evaluating a large number of potentially irrelevant concepts, AUROC is unsuitable, because it admits concepts that have generally low similarity, but spuriously separate the sets. However, in the case of propose-then-rank, all concepts are relevant to the images because the MLLM proposes them based on the seen images, so the likelihood of introducing noise is low. We denote the AUROC score of a candidate s as $\text{AUC}_A(s)$ when treating A as the positive class, and $\text{AUC}_B(s)$ when treating B as the positive class. Then, we select the top-N candidates by AUC_A as the final feature set $\widehat{\mathcal{S}}_A$ for set A, and the top-N candidates by AUC_B as $\widehat{\mathcal{S}}_B$ for set B (we use $N = 10$ unless stated otherwise).

D.2 Distribution Shift Explanations

The distribution shift baselines aim to discover the semantic differences between sets A and B by first identifying geometric directions in the image embedding space that capture the shift, and then decoding these directions into natural language concepts. This process consists of two stages: (i) subspace discovery in the joint embedding space, and (ii) vocabulary mapping and aggregation.

Let $\phi(x), \phi(y) \in \mathbb{R}^D$ denote the L2-normalized SigLIP2 image embeddings for images $x \in A$ and $y \in B$. We seek a set of orthogonal directions $u_c \in \mathbb{R}^D$ that best separate the two distributions. We evaluate two methods for finding these directions: Linear Support Vector Classification (LinearSVC) and U-WaX (Naumann et al., 2026).

Subspace Discovery. For the **Linear** approach, we iteratively discover a set of orthogonal directions. In each step c, we train a LinearSVC (with A as the positive class and B as the negative class) on the current embeddings to obtain a weight vector w_c. We orthogonalize w_c against all previously found directions using Gram-Schmidt orthogonalization, and L2-normalize it to obtain u_c. We then project all embeddings onto the orthogonal complement of u_c, i.e., $\phi(x) \leftarrow \phi(x) - (u_c^\top \phi(x))u_c$, and repeat the process to find the next direction.

For the **U-WaX** approach, we frame the shift as an optimal transport problem. We compute the pairwise squared L2 cost matrix between all $\phi(x)$ and $\phi(y)$, and solve for the optimal unregularized transport plan γ^* that couples instances from A and B. U-WaX then extracts orthogonal directions u_c that maximize the Wasserstein distance attributed to these specific subspaces, effectively isolating the sub-shifts that contribute most to the structural displacement between the sets.

Vocabulary Mapping and Aggregation. Once the shift directions $\{u_c\}$ are identified, we map them back to interpretable text concepts. Given a large vocabulary of candidate concepts $\mathcal{S}$ and their L2-normalized text embeddings $\psi(s)$, we compute the projection of each concept onto the shift direction via $\psi(s)^\top u_c$. We use the same list of 54,000 COCA terms introduced in Section 4.

Because the directions are oriented such that positive values indicate alignment with set A and negative values indicate alignment with set B, concepts with the highest positive scores are retrieved as candidate features for A. Conversely, concepts with the lowest negative scores (i.e., highest alignment with $-u_c$) are retrieved as candidate features for B.

D.3 Tagging and Aggregation Methods

The tagging and aggregation baselines operate by extracting textual representations from each image individually and subsequently summarizing these representations using an LLM. This approach aims to bypass the direct evaluation of image embeddings by mapping images to text first, thereby reframing the dataset difference problem as a text summarization task over discrete image attributes.

[1] For SigLIP models, we additionally apply sigmoid: $f(s, x) = \sigma\big(\exp(\ell) \cdot \phi(x)^\top \psi(s) + b\big)$, where σ is the sigmoid function, and ℓ and b are the learned logit scale and bias parameters of SigLIP2.

These baselines consist of two distinct stages: (i) per-image tagging, where textual features or captions are extracted for all images $x \in A$ and $y \in B$, and (ii) aggregation, where the collected textual data is summarized to identify the distinguishing concepts between the sets.

Per-Image Tagging. We evaluate three variants for extracting textual representations from the images, differing in the format and the model used:

1. **Caption Summary:** For each image, we use an MLLM (Gemma4-31B) to generate a detailed, long-form caption. This provides a holistic but unstructured description of the image content.
2. **Image Tagging (MLLM):** We prompt the MLLM to return a structured JSON output containing a list of all identified objects and their visual attributes (e.g., color, texture, shape, state). This potentially isolates atomic concepts better than free-form captions.
3. **Image Tagging (RAM++):** We use the RAM++ model (Zhang et al., 2024) to extract a set of predefined visual concepts (tags) present in each image. Because RAM++ operates on a fixed vocabulary, its outputs are intrinsically normalized across the dataset.

Aggregation. For the **Caption Summary** and **Image Tagging (MLLM)** variants, the extracted texts are highly variable. We address this by partitioning the captions into smaller subsets and processing them via repeated queries to an LLM. Specifically, we randomly sample small batches of captions from set A and set B. For each batch, we prompt an LLM (DeepSeek V4 Flash) to infer the semantic theme that distinguishes the two sets and to propose 10 characteristic features for A and 10 for B. This process is repeated across multiple independent queries to ensure broad coverage. Finally, all the proposed feature lists are concatenated and passed to the LLM in a consolidation prompt, which synthesizes the repeated observations into a final, representative list of 10 distinguishing features for set A and 10 for set B.

For the **Image Tagging (RAM++)** variant, the aggregation is more straightforward due to the fixed vocabulary. Let $\mathcal{T}$ denote the set of all unique tags identified across both datasets. For each tag $t \in \mathcal{T}$, we compute its frequency in set A and set B, denoted as $p_A(t)$ and $p_B(t)$. We then define the margin for a tag as the difference in proportions: $\delta(t) = p_A(t) - p_B(t)$. The top 10 tags with the highest positive margins are selected as the characteristic features for set A, and the top 10 tags with the lowest negative margins are selected for set B.

D.4 WORDLIST (COCA) + SIGLIP OR FG-CLIP EMBEDDINGS

In this paper, we use SigLIP2 image embeddings and FG-CLIP patch-level embeddings to discover visual concepts that distinguish sets A and B. FG-CLIP operates on localized image patches, while SigLIP operates on global representations, so FG-CLIP can capture small objects and attributes that might be overlooked by whole-image embeddings. In both cases, the method includes two steps: (i) image embedding and concept scoring, and (ii) vocabulary ranking based on concept score margins.

Image Embedding and Concept Scoring. Given images $x \in A$ and $y \in B$, we use SigLIP2 (Tschannen et al., 2025) or FG-CLIP2 (Xie et al., 2026) to extract image embeddings. In the case of SigLIP, the image encoder directly yields a single embedding vector $\phi(x)$. FG-CLIP instead returns a matrix of patch embeddings for each image. Here, $\phi(x) \in \mathbb{R}^{D \times \rho}$ is the matrix of L2-normalized patch embeddings for image x, where ρ is the number of patches and $\phi_i(x) \in \mathbb{R}^D$ is the embedding of the i-th patch.

We evaluate these images against a large pool of candidate text concepts $\mathcal{C}$, for which we compute L2-normalized text embeddings $\psi(s) \in \mathbb{R}^D$ using the SigLIP or FG-CLIP text encoder. As with the other methods, we use a vocabulary derived from the COCA corpus containing $M = 54,000$ concepts. To determine the relevance of a concept s to an image x, we compute the similarity score $f_{\text{emb}}(s, x)$. For SigLIP, this is calculated as

$$f_{\text{emb}}(s, x) = \sigma(\exp(\ell) \cdot \psi(s)^T \phi(x) + b), \tag{5}$$

where σ is the sigmoid function, and ℓ and b are the learned logit scale and bias parameters of the SigLIP model. For FG-CLIP, we compute the max-over-patches cosine similarity:

$$f_{\text{emb}}(s, x) = \max_{i \in \{1,\dots,\rho\}} \psi(s)^\top \phi_i(x). \tag{6}$$

This formulation ensures that a concept is scored highly if it aligns strongly with any local region in the image. We compute f_{emb} for all candidate concepts across all images in both A and B.

Vocabulary Ranking. After scoring all images against the vocabulary, we aggregate the scores to find the concepts that best distinguish the two datasets. For each concept s, we compute a divergence $\hat{\delta}_{A,B}(s)$ between the score distribution for set A and the score distribution for set B. This divergence can take multiple forms. In Section 4, we use the difference of means:

$$\hat{\delta}_{A,B}(s) = \mathbb{E}_{x\sim A}[f_{\text{emb}}(s,x)] - \mathbb{E}_{y\sim B}[f_{\text{emb}}(s,y)] \tag{7}$$

$$= \frac{1}{|A|}\sum_{x\in A} f_{\text{emb}}(s,x) - \frac{1}{|B|}\sum_{y\in B} f_{\text{emb}}(s,y), \tag{8}$$

which is easy to calculate and effective. In Section 5, we additionally divide the difference of means by the standard deviation of the concept's scores over both sets, i.e., we use a standardized effect size:

$$\overline{f}(s) = \frac{1}{|A|+|B|}\sum_{x\in A\cup B} f_{\text{emb}}(s,x) \tag{9}$$

$$\sigma(s) = \sqrt{\frac{1}{|A|+|B|}\sum_{x\in A\cup B}\left(f_{\text{emb}}(s,x) - \overline{f}(s)\right)^2} \tag{10}$$

$$\hat{\delta}_{A,B}(s) = \frac{1}{\sigma(s)}\left(\frac{1}{|A|}\sum_{x\in A} f_{\text{emb}}(s,x) - \frac{1}{|B|}\sum_{y\in B} f_{\text{emb}}(s,y)\right). \tag{11}$$

This makes concepts whose scores have very different spreads comparable.

In Section 6, we also use AUROC as divergence:

$$\hat{\delta}_{A,B}(s) = \frac{1}{|A||B|}\sum_{x\in A}\sum_{y\in B}\Bigg[\mathbb{I}\left(f_{\text{emb}}(s,x) > f_{\text{emb}}(s,y)\right) + \frac{1}{2}\mathbb{I}\left(f_{\text{emb}}(s,x) = f_{\text{emb}}(s,y)\right)\Bigg], \tag{12}$$

where $\mathbb{I}(\cdot)$ is the indicator function. The AUROC estimates the probability that a randomly chosen image from set A receives a higher concept score than a randomly chosen image from set B.

We then rank the concepts based on the score margin $\hat{\delta}_{A,B}(s)$. The concepts with the largest positive margins represent features strongly associated with set A relative to B, while those with the largest negative margins (i.e., highest $\hat{\delta}_{B,A}(s)$) represent features characteristic of set B.

MLLM Optimization. To discover specific visual differences that are not captured in the static wordlist, such as rare concepts and multiword expressions, we optionally perform an iterative refinement of the candidate concepts using an MLLM. We initialize the optimization with the top-ranked concepts (by score margin $\hat{\delta}_{A,B}(s)$) from the wordlist.

At each iteration, the MLLM is provided with an image grid containing examples from both set A and set B, see Fig. 10 for an example. Alongside this visual context, the MLLM is given a historical record of currently active and previously discarded textual concepts, along with their discriminative metrics (such as the difference of means and AUROC).

Prompted with this multimodal context, the MLLM infers the underlying semantic themes and proposes novel, semantically distinct candidate phrases to better separate the two sets. We then compute the text embeddings $\psi(s)$ for these new proposals, evaluate their similarity scores $f_{\text{emb}}(s,x)$ against the image sets, and calculate their score margins $\hat{\delta}_{A,B}(s)$. After filtering out semantically redundant phrases, the newly proposed concepts are pooled with the existing candidates, and we retain only the top-k most discriminative concepts for the next iteration.

Combining SigLIP and FG-CLIP Scores. To use the complementary strengths of global and local representations, we combine the scores from both SigLIP and FG-CLIP to rank concepts. Because

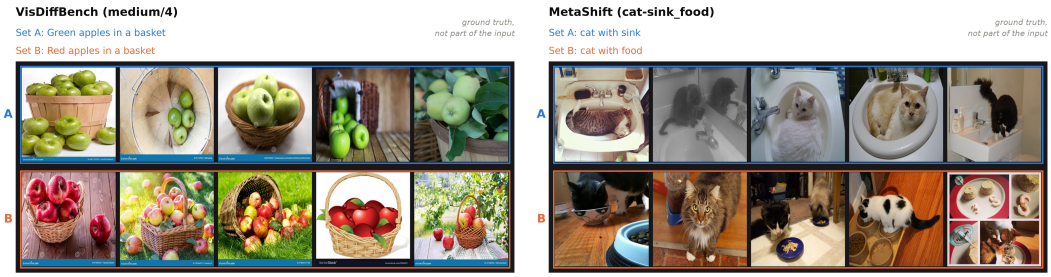


Figure 10: Example visual input grids provided to the MLLM during optimization. Each grid displays sample images from Set A (top row, blue) and Set B (bottom row, orange) for tasks from VisDiffBench (left) and MetaShift (right). Ground-truth descriptions above each grid are shown for reference only and are not included in the model's input.

the raw similarity scores $f_{\text{emb}}(s, x)$ from the two models are on different scales and follow different distributions, we first map them to a common scale before comparison. To achieve this, we calibrate the cosine similarities of each model to presence probabilities $p(s, x) = \sigma\left(a \cdot \psi(s)^\top \phi(x) + b\right)$, where a and b are fitted once per model by logistic regression on the object-presence labels of SUN RGB-D (Platt scaling). For FG-CLIP, $\psi(s)^\top \phi(x)$ is the max-over-patches similarity. After calibration, we calculate the divergence $\hat{\delta}_{A,B}(s)$ as the difference of mean presence probabilities in both SigLIP space and FG-CLIP space, which are now directly comparable. Then, we take the maximum divergence score over the two spaces, i.e., formally, the combined divergence is defined as

$$\hat{\delta}_{A,B}(s) = \max\left(\hat{\delta}^{\text{SigLIP}}_{A,B}(s), \hat{\delta}^{\text{FG-CLIP}}_{A,B}(s)\right). \tag{13}$$

By taking the maximum, a concept is deemed highly discriminative if it effectively separates the two image sets in either the global representation or the fine-grained view.

E Technical Details of the Multi-Target Evaluation

E.1 How we Sample the Multi-Target Datasets

To simulate complex distributions with overlapping concepts, we cannot sample images for each concept independently because multi-object co-occurrence (in one image) entails that assigning an image to a set to satisfy the demand for one concept will at the same time change the counts of all other concepts present in that image. To address this challenge, we frame multi-target dataset construction as an optimization problem over the assignments of all images.

Target Definition. Let n be the target size for each set (A and B). For every concept c in the dataset (e.g., COCO or SUN RGB-D classes), we assign a role (aligned to A, aligned to B, or neutral), an alignment ratio r_c, and an overall prevalence p_c. The overall prevalence is identical to the prevalence of the concept in the source dataset (e.g., COCO, LVIS, or SUN RGB-D), with a random perturbation that allows up to $\pm 10\%$ deviation from the original prevalence. From the roles, alignment ratios r_c, and prevalence r_c, we compute count targets t_c^A and t_c^B for Set A and Set B. For example, if a concept is aligned to Set A, its count target satisfies $\frac{t_c^A}{t_c^A + t_c^B} \geq r_c$.

Objective Function. We define a state where each image i is assigned to Set A, Set B, or excluded. Let c_c^A and c_c^B be the realized counts of concept c in the current assignment, and let $|A|$ and $|B|$ be the current sizes of the sets. We minimize a loss function $\mathcal{L}$ that penalizes squared deviations from both the concept targets and the overall set size limits:

$$\mathcal{L} = \sum_c \left[(c_c^A - t_c^A)^2 + (c_c^B - t_c^B)^2\right] + \lambda\left[(|A| - n)^2 + (|B| - n)^2\right] \tag{14}$$

where $\lambda = 1.0$ balances the concept counts with the set size constraints.

Initialization (Greedy Assignment). We greedily initialize the sets A, B, and excluded images. For each image, we calculate the unmet demand in Set A and Set B for the concepts it contains. The

image is assigned to the set with the highest unmet demand, provided that set has not yet reached the maximum capacity n. If neither set needs the image's concepts, it is excluded.

Iterated Local Search. The greedy initialization provides a strong starting point but is sub-optimal due to complex co-occurrences which cannot be satisfied greedily. To refine the sets, we apply an iterated local search: In each step, we randomly select an image and evaluate the loss $\mathcal{L}$ for all three of its possible states (assigned to A, assigned to B, or excluded), keeping the assignments of all other images fixed. The image is reassigned to the state that strictly minimizes the global loss. This process trades off local shortfalls against competing per-class targets and is repeated until convergence (i.e., when no improvements are found after 20,000 iterations).

E.2 Multi-Level Masked Concept Selection

Here, we explain how we use multi-level masked concept selection to select more diverse concepts from the overall pool (MLLM-proposed or from a static wordlist). Let $\mathbf{S} \in \mathbb{R}^{(|A|+|B|)\times P}$ be the score matrix evaluated over all images in sets A and B, where the rows are indexed by the images and the columns are indexed by the P candidates in the pool (e.g., the COCA word list). Let $\mathbf{y} \in \{0,1\}^{|A|+|B|}$ indicate set membership (in A or in B).

Per-Image Top-K Masking. For a given threshold level K, we define the masked score matrix $\mathbf{S}^{(K)}$ by retaining, in each row, only the K highest-scoring candidates for that specific image:

$$\mathbf{S}^{(K)}_{i,j} = \begin{cases} \mathbf{S}_{i,j} & \text{if } j \in \operatorname{top}_K(\mathbf{S}_{i,:}), \\ 0 & \text{otherwise,} \end{cases} \tag{15}$$

where $\operatorname{top}_k(\mathbf{S}_{i,:})$ is the index set of the k largest entries in the i-th row. This procedure suppresses candidates that are diffusely elevated across many images without ever being a strong match for any single image, while preserving genuine concepts that are strongly present in the image.

Per-Level Selection. For each masking level $K \in \mathcal{K} = \{10, 50, 100\}$, we compute the score divergence from the masked matrix:

$$\delta^{(K)}(s_j) = \operatorname{sign} \cdot \left(\frac{1}{|A|} \sum_{i \in A} \mathbf{S}^{(K)}_{i,j} - \frac{1}{|B|} \sum_{i \in B} \mathbf{S}^{(K)}_{i,j} \right). \tag{16}$$

Here, we use the standardized image-concept similarity scores. We then apply a selection method g (such as Top-k or Residual-OMP) to the masked matrix and divergence scores and get an ordered list of 10 candidates:

$$L^{(K)} = f(\mathbf{S}^{(K)}, \delta^{(K)}) = \left(c_1^{(K)}, \ldots, c_{10}^{(K)}\right). \tag{17}$$

We run this independently for each $K \in \mathcal{K}$, which yields three rank-ordered lists $L^{(10)}, L^{(50)}$, and $L^{(100)}$ per target side (A and B).

Round-Robin Aggregation. Finally, the three candidate lists are merged into a single list by round-robin interleaving over ascending K and rank r:

$$\left(c_1^{(10)}, c_1^{(50)}, c_1^{(100)}, c_2^{(10)}, c_2^{(50)}, c_2^{(100)}, \ldots\right). \tag{18}$$

This combined list is deduplicated by keeping only the first occurrence of each concept and then truncated to 10 concepts, so a candidate concept that is re-selected at a later level does not consume an additional slot, and we return exactly 10 concepts.

E.3 Residual Orthogonal Matching Pursuit (Residual-OMP)

In Section 5, we find that in complex and noisy datasets, naive top-k selection of concepts based on the score divergence is suboptimal, because the top-ranked concepts are often redundant or repetitive. Therefore, we propose and evaluate orthogonal matching pursuit (OMP) as a method to select a subset of k high-scoring concepts that is also diverse. Concretely, the **Residual-OMP** (Orthogonal Matching Pursuit) method selects a subset of concepts by greedily capturing the variance in the target binary labels that remains unexplained by previously selected concepts.

Let $\mathbf{S} \in \mathbb{R}^{(|A|+|B|)\times M}$ be the matrix of concept scores evaluated over all images in sets A and B, where $S_{i,j} = f_{\text{emb}}(s_j, x_i)$. We first standardize the columns (concepts) of $\mathbf{S}$ such that each concept

has zero mean and unit variance across the dataset, yielding the normalized design matrix $\mathbf{X}$. Let $\mathbf{y} \in \{0,1\}^{|A|+|B|}$ be the binary indicator vector where $y_i = 1$ if $x_i \in A$ and 0 otherwise.

At the beginning of the selection process, we initialize the residual target vector as $\mathbf{r}^{(0)} = \mathbf{y}$ and the set of selected concept indices as $\mathcal{K} = \emptyset$. At each iteration $t \in \{1, \ldots, k\}$ (where k is the target budget), we select the concept $j^* \notin \mathcal{K}$ that maximizes the absolute correlation with the current residual:

$$j^* = \arg\max_{j \notin \mathcal{K}} \left| \mathbf{X}_{:,j}^\top \mathbf{r}^{(t-1)} \right|. \tag{19}$$

We then update the active set $\mathcal{K} \leftarrow \mathcal{K} \cup \{j^*\}$. Next, we orthogonalize the target vector against the subspace spanned by all selected concepts. Let $\mathbf{X}_\mathcal{K}$ be the submatrix of $\mathbf{X}$ containing only the columns in $\mathcal{K}$. We fit a linear model to predict $\mathbf{y}$ using $\mathbf{X}_\mathcal{K}$ (via least squares) and compute the new residual:

$$\mathbf{r}^{(t)} = \mathbf{y} - \mathbf{X}_\mathcal{K}(\mathbf{X}_\mathcal{K}^\top \mathbf{X}_\mathcal{K})^{-1} \mathbf{X}_\mathcal{K}^\top \mathbf{y}. \tag{20}$$

This process is repeated until k concepts are selected. By refitting the linear model on the full active set and updating the residual at each step, Residual-OMP naturally penalizes redundancy, selecting concepts that provide orthogonal discriminative signal rather than merely restating information captured by earlier selections.

E.4 Detailed Description of Metrics for Multi-Target Evaluation

In Section 5, we evaluate methods using three metrics that focus on ground-truth concept recall from different perspectives:

MCC-R measures whether the selected concepts, together with their similarity scores f_{emb}, can predict the binary ground-truth concept membership. An aligned ground-truth concept c counts as recovered if and only if one of the K predicted concepts reaches a Matthews Correlation Coefficient (MCC) of at least $\tau = 0.5$ against the binary class membership labels, for any threshold on the continuous f_{emb} image-concept scores, i.e., for each concept we take the maximum over all thresholds $\max_t \mathrm{MCC}\big(\mathbf{1}[f_{\text{emb}(s)} \geq t],\ y_c\big)$ where y_c are the binary class memberships. For each task, we calculate the ratio of ground truth concepts that can be recovered in this way as a recall metric.

In general, the MCC for a specific binarization threshold t is defined as:

$$\mathrm{MCC}(t) = \frac{\mathrm{tp}(t)\,\mathrm{tn}(t) - \mathrm{fp}(t)\,\mathrm{fn}(t)}{\sqrt{(\mathrm{tp}(t) + \mathrm{fp}(t))\,(\mathrm{tp}(t) + \mathrm{fn}(t))\,(\mathrm{tn}(t) + \mathrm{fp}(t))\,(\mathrm{tn}(t) + \mathrm{fn}(t))}}, \tag{21}$$

where $\mathrm{tp}(t)$ are the true positives when evaluating the ground-truth per-image class memberships against the binarized score memberships, and equivalently for tn (true negatives), fp (false positives), fn (false negatives). Intuitively, MCC quantifies the overall correlation between the induced binary classifications and the ground-truth binary labels across all four quadrants of the confusion matrix (tp, tn, fp, fn), which provides a balanced score even when the classes are heavily imbalanced. This is the case for many aligned classes in our setup that have low overall prevalence.

We use MCC-R to provide a holistic understanding of how well a method, which includes the concept selection and the image-concept scores, can reconstruct the structure of the dataset, i.e., the aligned concepts and their per-image presences. This is a realistic assessment because in deployment, a user would only have access to the predicted concepts and their scores, which must be used to reason about the data.

One arbitrary design choice is the threshold $\tau = 0.5$ that indicates when we count a concept as recovered. Therefore, we ablate the effect of τ and sweep the range of $\tau \in \{0.2, 0.25, \ldots, 0.8\}$. Results are in Fig. 11. We find that, as expected, MCC-R decreases with increasing τ, as it becomes harder to confirm a concept. While the absolute value of the metric changes with τ, the ordering of methods is stable up to the extremes, i.e., the choice of τ does not influence the order and relative gains of methods. The comparability of MLLM-proposed concepts and COCA concepts also holds true across thresholds.

Embedding Space Cosine Similarity. As in Section 4 (Appendix A.1), we evaluate the concept recall by embedding both predicted and ground-truth concepts with the all-MiniLM-L6-v2 sentence embedding model. Then, for every ground-truth concept, we find the maximum cosine similarity across predicted concepts, and finally average the maximum similarities across concepts.

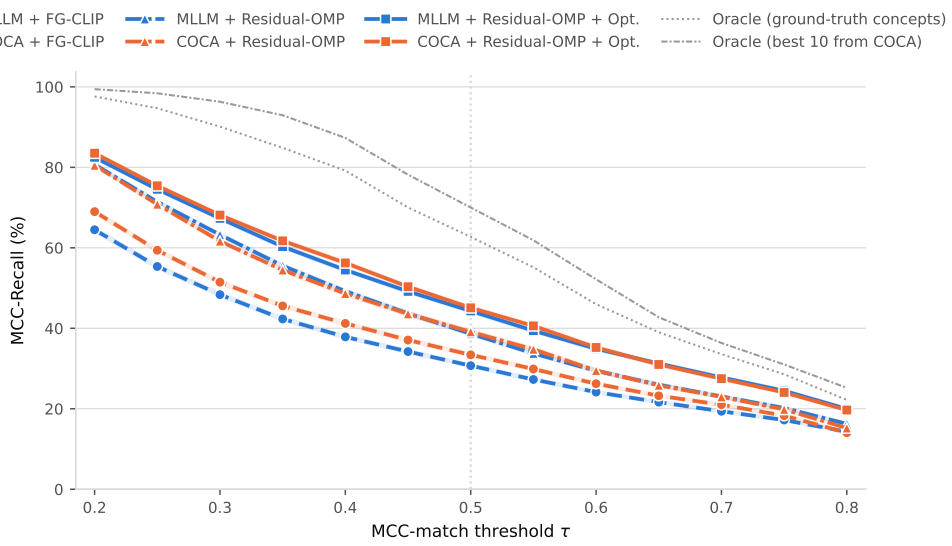


Figure 11: Sensitivity analysis over the MCC-match threshold τ. MCC-Recall (%) across varying recovery thresholds $\tau \in [0.2, 0.8]$ for different concept discovery and scoring methods compared against oracle baselines. The vertical dotted line denotes the default threshold $\tau = 0.5$ used in the main experiments.

LLM-Judged Recall. We also measure recall by using an LLM to judge concept equivalence (metric R_J). In this case, we provide the list of ground-truth concepts and the list of predicted concepts to the LLM. The LLM then returns a list of all ground-truth concepts that have a related concept among the predicted concepts. Here, we define "related" as semantically equivalent, synonymous, or strongly associated. We also accept implied concepts, for example "stuffed animal" $\rightarrow$ "teddy bear". The following (real) example illustrates how this metric is calculated:

Ground-Truth:	apple, backpack, car, cow, knife, **sheep**, **teddy bear**
Predicted:	**toy bear**, **teddy bear**, wide open grassy field, pastoral scenery, livestock animal, rural setting, **sheep in a field**, **stuffed animal**, pasture field, **farm animal**
Matched:	teddy bear, sheep

The LLM finds two (of 7) ground-truth concepts in the predicted concepts. Both ground truth concepts are matched with multiple predicted concepts each, and in the case of "sheep", no predicted concept is identical to the ground-truth concept name. In this case, the recall $R_j = \frac{2}{7}$.

For the results in Table 2, we use Qwen3.8-27B as an LLM, but also calculate all metrics with gpt-oss-120B for comparison. The results are in Fig. 12. We find that scores differ slightly between Qwen3.8-27B and gpt-oss-120B, but not substantially, and they never flip the relative performance of any two methods, i.e., which one performs better.

Overall, we find that all three recall metrics largely agree with each other. Therefore, finally, we also analyze discriminative metrics on the setting in Section 5. In Section 4, discrimination (D) is a useful metric because the target concept is present in almost all images, and all methods perform well. However, in the setting in Section 4, the prevalence of target concepts is much lower. In consequence, not even ground-truth concepts achieve meaningful separation between sets A and B. Instead, we can only detect which concepts are more prevalent in set A or in set B, but we cannot necessarily use them anymore to globally distinguish the sets.

To show this, we evaluate three alternative metrics on the same methods as in Section 5, i.e., in Table 2. The alternative metrics are defined as follows:

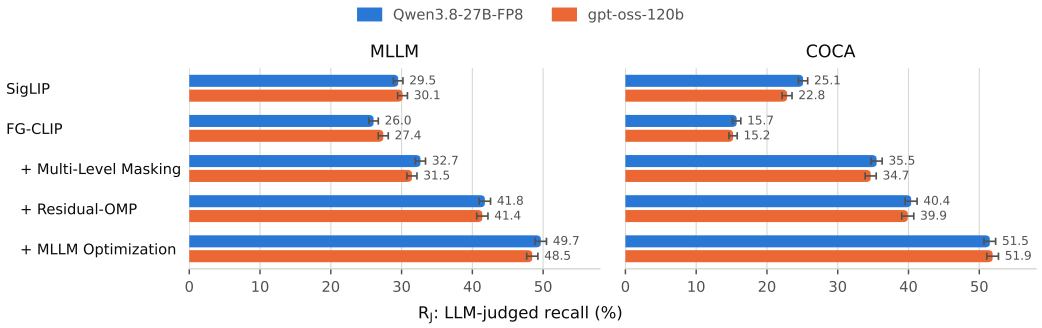


Figure 12: LLM Judge Recall R_J evaluated with Qwen3.8-27B and gpt-oss-120b. Scores are averaged over datasets. Error bars show 95% CIs.

Set-Level Discrimination ***(D).*** This metric trains a logistic regression classifier on the continuous f_{emb} image-concept scores for the predicted concepts in 5-fold cross-validation. The classifier receives as input the 20 scores between an image and the concepts predicted for each set. From this, the classifier predicts the set membership of the image, i.e., A or B.

Concept-Level Discrimination ***(D_c).*** This metric works similarly to Set-Level Discrimination (D), but instead of predicting the set membership, the classifier predicts the presence or absence of ground-truth classes (or concepts) in the image. Specifically, for each set (A and B), we train a logistic regression classifier in 5-fold cross-validation using only the continuous f_{emb} scores of the 10 concepts predicted for that specific set. The classifier's target is the binary ground-truth presence of each planted class. We evaluate the performance using the out-of-fold Area Under the Receiver Operating Characteristic (AUROC) rather than accuracy due to varying class prevalences, and average the scores over all ground-truth classes.

Concept-Level Binarized Discrimination ***(D_b).*** This metric is identical to Concept-Level Discrimination (D_c), except that the continuous f_{emb} scores are binarized before training the classifier. For each predicted concept, we first find the ground-truth class it best matches and determine the threshold that maximizes the MCC for that match. The continuous scores for that concept are then binarized (0 or 1) using this specific threshold. We then train the same logistic regression classifier in 5-fold cross-validation on this binary matrix to predict the ground-truth class presences, again reporting the out-of-fold AUROC. While MCC-R evaluates if single concepts pass a strict threshold, D_b measures whether the full set of predicted concepts, once reduced to binary indicators, can jointly decode the ground-truth classes.

Results are in Table 9. We make the following observations: Discrimination (D) often yields scores that are only barely better than chance (50.0), including for ground-truth concepts. Only for DOTA, scores are better, reaching around 70%. Furthermore, scores are inconsistent with the recall scores, which were generally lower for SUN RGB-D, but are higher for discrimination. We think the main reason for these problems is that the setting in Section 5 does not admit easy discrimination of sets because target concepts have overall low prevalence, and they are not entirely exclusive to either set. Another problem is fitting a classifier, which can overfit on spurious associations and use signals from all features combined instead of isolating individual concepts. Concept-Level discrimination metrics (D_c and D_b) to some extent inherit this problem, but they agree better with recall metrics and generally show small variation between methods. Overall, we find using discrimination metrics difficult in this setting, and therefore focus on recall metrics.

F Technical Details of Applications

F.1 Workflow for Finding Spurious Correlations in ImageNet

To find spuriously correlated class-concept pairs in ImageNet, we proceed in three steps: (1) We determine a candidate pool of concepts where we search for spurious correlations with classes. (2) We filter the full product of classes $\times$ concepts using a number of tests, which involve finding

	LVIS				MS COCO				SUN RGB-D				DOTA			
	MCC-R	D	D_c	D_b	MCC-R	D	D_c	D_b	MCC-R	D	D_c	D_b	MCC-R	D	D_c	D_b
RAM++	31.5	57.8	72.8	72.6	44.7	60.4	76.9	76.4	15.1	60.5	68.4	68.2	29.4	66.2	75.9	75.0
MLLM + SigLIP	26.4	54.7	80.6	68.8	40.2	57.3	87.6	73.0	10.1	58.7	74.8	66.8	33.8	70.9	87.4	74.6
MLLM + FG-CLIP, no masking	22.2	56.5	78.6	70.3	27.4	58.8	84.1	73.4	15.8	61.5	72.8	66.5	31.1	69.3	83.3	74.1
MLLM + FG-CLIP	27.5	54.9	80.5	72.2	42.4	57.5	87.8	77.6	20.6	60.8	75.2	69.9	32.4	68.8	83.3	74.3
+ Residual-OMP	32.8	56.2	**82.9**	**78.1**	**53.5**	58.9	**90.7**	**84.5**	**22.9**	61.7	**77.3**	73.2	45.5	70.9	86.5	82.4
COCA-20k + SigLIP	24.3	54.9	79.5	69.2	40.7	57.4	87.1	74.8	6.6	59.0	75.3	67.9	34.9	71.5	**88.4**	79.1
COCA-20k + FG-CLIP, no masking	16.5	**59.2**	77.2	70.0	17.8	**60.9**	82.5	72.1	11.3	**63.5**	71.2	67.1	28.2	**71.6**	82.7	76.7
COCA-20k + FG-CLIP	30.0	55.7	81.2	74.1	46.8	58.2	89.0	80.5	21.4	60.8	75.8	71.7	35.5	69.4	84.1	77.2
+ Residual-OMP	**33.4**	56.7	82.7	77.6	52.7	59.1	90.5	84.4	22.6	61.4	77.0	**73.7**	**47.7**	**71.6**	86.5	**82.5**
Oracle (ground-truth concepts)	59.4	55.4	91.9	84.4	96.3	57.1	97.4	88.3	36.8	60.3	84.3	77.9	58.4	67.7	83.7	78.8

Table 9: Comparison of MCC-Recall and alternative discrimination metrics across datasets. We report Set-Level Discrimination (D), Concept-Level Discrimination (D_c), and Concept-Level Binarized Discrimination (D_b) on LVIS, MS COCO, SUN RGB-D, and DOTA. Bold and underlined values denote the best and second-best performing methods.

surprisingly frequent associations, removing class-concept pairs where the class and concept are semantically closely related, and sorting candidates by likelihood and interestingness of potential correlations. (3) Finally, we verify a shortlist of 6,103 class-concept pairs with an MLLM.

Concept Candidates. Our initial pool of candidates for spurious correlations is a subset of the COCA wordlist (Davies, 2010). We take the 20,000 most frequent words, and for each word, we ask an LLM (Gemma4-31B in our case) how visually concrete the word is. The LLM assigns an integer score between 1 and 10 to each word, and we instruct it with the following examples:

"sunglasses"	$\rightarrow 10$	(specific physical object)
"torn"	$\rightarrow 9$	(directly visible material state)
"sponsored"	$\rightarrow 6$	(partially concrete, grounded in visible logos)
"confrontation"	$\rightarrow 3$	(occasionally visually, e.g., two people facing off)
"predator"	$\rightarrow 3$	(behavioral role, not visual feature)
"conservation"	$\rightarrow 1$	(a policy/topic, not something a photo shows)

This instruction is deliberately biased toward detecting concrete class-object and class-attribute correlations, and not more abstract themes. We acknowledge that investigating this is of interest too, but abstract concepts are harder to verify, especially with automatic methods. After obtaining the scores from Gemma4-31B (see Fig. 13, left panel, for their distribution), we keep all concepts with concreteness score ≥ 8, which leaves 5,317 concepts.

Filtering Class-Concept Pairs. Next, we filter the full product of 1000 classes $\times$ 5,317 concepts. Concretely, we drop pairs where the embedding cosine similarity of class name and concept is ≥ 0.7 using the EmbeddingGemma-300m model (Vera et al., 2025). Also, we calculate the FG-CLIP similarity scores $f_{\text{emb}}(s, x)$ between all concepts and images. For each image, we only keep the scores of the 100 top ranking concepts by z-score, i.e., the 100 concepts with the highest score, and set all other scores to 0. Here, we first standardize concept similarity scores based on their per-concept mean and standard deviation across all images before comparing them. Based on this, we remove class-concept pairs where the concept has a nonzero score for < 20 images of the class.

Using the sparsified scores, we conduct two tests for each class-concept pair. First, we test whether the concept can separate the class from all other classes (one-vs-rest) with AUROC at least 0.55. In our formulation, AUROC is the rank-sum Mann-Whitney U statistic:

$$\text{AUROC}(c, s) = \frac{R_{c,s} - 0.5 \cdot n_c(n_c + 1)}{n_c \cdot (N - n_c)}, \tag{22}$$

N is the total number of images in the target and background sets combined, n_c is the number of images in class c, and $R_{c,s}$ is the summed rank of the images in class c when sorting all ImageNet images by the concept similarity f_{emb}. Note that here, higher scoring images have *higher* rank. We then keep class-concept (c-s) pairs where $\text{AUROC}(c, s) \geq 0.55$. Additionally, we keep a maximum of 200 concepts with the highest AUROC per class at this stage. Overall, this step finds class-concept pairs where the concept appears more frequently in images of the class than expected, using the entire ImageNet as the background distribution. Note that this is an instance of image dataset diffing.

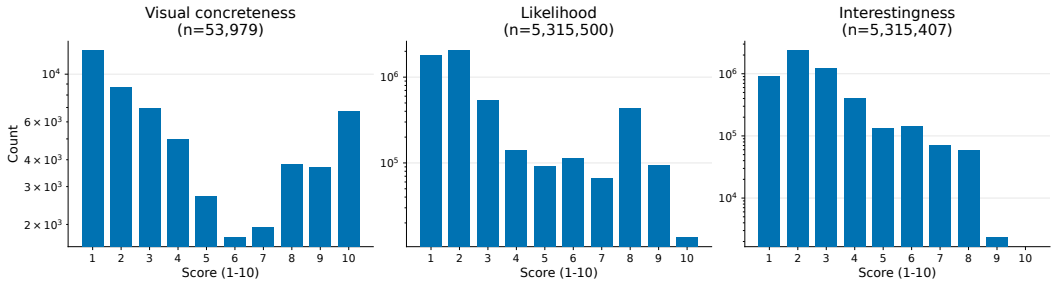


Figure 13: Distribution of LLM-assigned filtering scores. Histograms show the distributions of integer scores (1 to 10) across candidate concepts and class-concept pairs. Note the logarithmic count scale. *Left:* Visual concreteness scores for concepts (Gemma4-31B). *Center:* Plausibility scores for class-concept pairs (Qwen3.8-27B). *Right:* Interestingness scores for class-concept pairs.

Next, we filter class-concept pairs with two scores representing interestingness and plausibility of the correlation. The purpose of this stage is to create a shortlist of candidate pairs for expensive MLLM verification. Pairs that survive the previous filtering steps are either genuinely correlated in ImageNet, or noise introduced by FG-CLIP biases, such as high visual or semantic similarity although the concept is not actually present. The interestingness score targets concepts that are genuinely correlated, but not unexpected. For example, it is not surprising that sheep often appear on pastures, or cars in urban areas. Hence, these combinations should receive low interestingness scores. The plausibility score targets noise and FG-CLIP biases. For example, pairs like hamster-aghast or wild_boar-dissolved are nonsensical and should be discarded. To get interestingness and plausibility scores, we ask an LLM (Qwen3.8-27B in our case) with class-concept pairs and appropriate elicitation prompts to provide integer scores between 1 and 10. See Fig. 13 (center and right panels) for their distribution. To improve computational efficiency, we generate scores for 10 concepts in each inference step at once. Class-concept pairs need a minimum likelihood score ≥ 3 and minimum interestingness score ≥ 5 to pass this stage.

We then compare surviving concepts to their distribution in the 20 most similar classes by concept distribution. Here, we represent each class by the 5,317-dimensional vector whose components represent the mean concept similarity scores f_{emb} over images in the class. Similarity between classes is determined by cosine similarity of these vectors. We include this step because we noticed that many trivial associations, such as dog-collar, are not filtered by the previous test. Here, "collar" is indeed more frequent in dog classes than in ImageNet in general, but it is not a special or unexpected correlation. In these cases, we find similar trends in neighboring classes. We calculate $\text{AUROC}(c, s)$ in the same way as Eq. (22), only changing the background corpus, i.e., replacing the entire ImageNet dataset with images from the 20 most similar classes. In this stage, the threshold a class-concept pair has to pass is $\text{AUROC}(c, s) \geq 0.55$, and we only keep the 15 highest scoring concepts per class.

MLLM Verification. To verify the shortlisted class-concept pairs, we prompt an MLLM (Qwen3.8-27B) to determine whether the concept s is clearly and visually present in images of the class c. Instead of exhaustively testing all images in a class, we use a batched verification process. For each candidate pair (c, s), we rank the images in class c in descending order by their FG-CLIP similarity score $f_{\text{emb}}(s, x)$. We then query the MLLM in batches, i.e., we first test the k highest-scoring images, then the next batch of k images, and so on. We use $k = 20$, but the first two tests are performed after 20 images, then 200 images, and then after each batch of 20 images. After each batch, we perform a binomial test (with Benjamini-Hochberg FDR control) to determine if the prevalence of the concept in the class is not lower than 10%. If we find that with the given confidence level (we use $\alpha = 0.05$) the candidate appears in less than 10% of images, we drop it early to reduce computational cost. We return all concepts that survive all batches, i.e., are tested on all images of the class and appear in at least 10% of them. In our case, this yields 1,553 class-concept pairs.

F.2 Workflow for Finding Demographic Biases in Text-to-Image Models

To systematically detect and verify demographic biases in text-to-image (T2I) models, we proceed in three steps: (1) We generate large-scale datasets of images using prompts with and without specific demographic descriptors to form contrast sets. (2) We extract and filter a candidate pool of biased concepts using vision-language models (FG-CLIP and SigLIP) coupled with Residual-OMP for diverse concept selection. (3) Finally, we verify the presence and statistical significance of these candidate biases using an independent MLLM.

Image Generation. We evaluate biases for country (10 categories), religion (5 categories), and race (7 categories) terms. For Flux 1-dev and Stable Diffusion 3.5 Large (as T2I models), we generate 100 images for every target prompt with template

> "a {country|religion|race adjective} {man|woman} {is working as occupation|doing activity}".

Here, we use 60 occupations and 45 common activities. This means that, per demographic target, we generate 10,500 images. For each prompt, we also generate a corresponding baseline set of 10,500 images using the exact same prompt but omitting the demographic target, i.e., the country/religion/race specifier. This pairs a target distribution (e.g., "an Egyptian woman") against a baseline distribution (e.g., "a woman"), which allows us to isolate visual concepts whose frequency is uniquely amplified or reduced by the country/religion/race term.

Finding and Filtering Bias Concepts. We reuse the candidate concept vocabulary from our ImageNet analysis (Appendix F.1), i.e., 5,317 visually concrete concepts from the COCA wordlist. Using these concepts, we first compute the image-text similarity scores $f_{\mathrm{emb}}(s, x)$ for all concepts across all generated images using both FG-CLIP and SigLIP models. To determine if a concept is disproportionately associated with the demographic term, we compare its score distributions in the target image set versus the baseline image set. We quantify the separation using the AUROC (Eq. (22)), i.e., the probability that a randomly chosen image from the target distribution (with demographic term specified) scores higher than a randomly chosen image from the baseline distribution (without demographic term). We conduct a two-tailed Mann-Whitney U test and apply Benjamini-Hochberg False Discovery Rate (FDR) control across all concepts to confirm candidates.

To ensure diversity in the final candidate list and reduce semantic redundancy, we select 30 concepts per prompt pair using the Multi-Level Masking and Residual-OMP methods proposed in Section 5. Due to the wealth of discovered attributes and the cost of additional optimization, we do not use MLLM optimization in this application.

MLLM Verification. We check the candidates using an MLLM (Qwen3.8-27B). Instead of testing concepts on all generated images (2×10500, computationally infeasible), we construct a stratified sample of 900 images per side (target and baseline prompt). We then ask the MLLM whether a concept is clearly visible in the image. To reduce inference cost, we query the MLLM with chunks of 10 concepts at a time. Finally, we perform the Fisher exact test on the MLLM's binary presence judgments to compare the prevalence of a concept in the target distribution against the baseline. Only candidates that show a statistically significant shift in visual presence that aligns with the direction predicted by the SigLIP or FG-CLIP embeddings are confirmed as genuine demographic biases. Additionally, we require that the prevalence in the target distribution differs from the baseline distribution by at least 10% according to MLLM presence counts.