GH-ESD: Grounded Hypothesis-Driven Error Slice Discovery for Instance-Level Vision Tasks

Wei Zhang[1⋆], Chaoqun Wang[1⋆], Zixuan Guan[1], Ping Sheng Kao[2], Pengfei Zhao[1], Peng Wu[1], and Sifeng He[1⋆⋆]

[1] Apple, Beijing, China
{wzhang52, chaoqun_wang, zixuan_guan, pzhao23, pwu4, he_sifeng}@apple.com
[2] Apple, Cupertino, CA, USA
pingsheng_kao@apple.com

Abstract. Systematic failures of vision models on semantically coherent subsets, known as error slices, reveal limitations in robustness and evaluation. Existing slice discovery approaches largely model slices as clusters in representation space or combinations of predefined attributes. While effective for image-level classification, such formulations are insufficient for instance-level tasks such as object detection and segmentation, where failures often arise from contextual, relational, and spatially grounded visual patterns. We propose GH-ESD (Grounded Hypothesis-Driven Error Slice Discovery), a generate-and-verify framework that reformulates slice discovery as grounded hypothesis generation and statistical verification. GH-ESD constructs relational failure hypotheses using LLM priors and grounded visual evidence, discovers hypothesis slices at the instance level via Vision-Language Models, and verifies them through statistical trend analysis over instance-level errors. We also introduce GESD (Grounded Error Slice Dataset), a new benchmark for instance-level error slice discovery, providing expert-defined and spatially grounded slices derived from detection and segmentation failures. Extensive experiments demonstrate that GH-ESD consistently outperforms baselines, improving Precision@10 by 0.10 (0.73 vs. 0.63) on the GESD benchmark for detection tasks, while also supporting segmentation scenarios. GH-ESD identifies interpretable slices that facilitate actionable model improvements.

Keywords: Error Slice Discovery · Vision-Language Models · Model Robustness · Object Detection and Segmentation Model Evaluation

1 Introduction

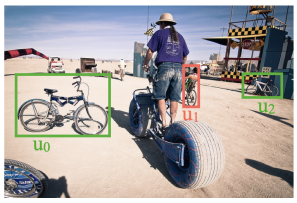
Despite the remarkable performance of modern deep vision models, their reliability remains challenged by systematic failures on semantically coherent subsets of data, commonly referred to as *error slices* [5]. These slices expose structured weaknesses in model behavior and provide critical signals for robustness analysis

⋆ These authors contributed equally to this research.
⋆⋆ Corresponding Author, Project Lead

Error Slice S: Bird misclassified as Waterway
Membership: Image ∈ S

Per-image error: 1

(a) Image-Level Formulation (NICO++)



Error Slice S: Bicycle partially occluded by person mis-detected
Membership: $u_0 \notin S$, $u_1 \in S$, $u_2 \notin S$

Image-level Aggregation:
Mean error = (0+1+0)/3 (diluted signal)

(b) Image-Level Aggregation of Multiple Instances

Instance-level errors:
$(u_0, u_1, u_2) = (0, 1, 0)$

(c) Instance-Level Formulation (GESD)

Dimension	Image-Level (Prior Work)	Instance-Level (Ours)
Evaluation Unit	Entire Image	Object Instance / Spatial Region
Failure Pattern	Spurious Correlations	Contextual & Relational Interaction
Error Signal Sensitivity	Diluted by multi-instance aggregation	Localized & Instance-level
Grounding	Implicit (Image-Level)	Explicit (Spatial Grounding)

Fig. 1: Comparison of image-level and instance-level error slice discovery.

and model refinement, particularly in high-stakes domains such as autonomous driving [2] and robotics [31]. Identifying such slices is essential for trustworthy model evaluation.

Most existing slice discovery methods implicitly model slices as regions in representation space or combinations of predefined attributes [4, 8, 9, 11, 12, 29]. While effective for surfacing spurious correlations in image classification (Fig. 1(a)), these global formulations are insufficient for instance-level tasks like object detection and segmentation, where failures are often driven by contextual, relational, and spatially grounded visual patterns (Fig. 1(c)). Recent efforts have extended slice discovery to detection tasks by clustering object embeddings [28] or generating object tags [5]. However, representing slices as embedding clusters or tag combinations may not always capture complex compositional semantics (e.g., contextual, spatial, and similarity relationships). Furthermore, these methods often rely on image-level error metrics for evaluation (Fig. 1(b)), which can restrict precise instance-level failure analysis.

To address these limitations, we propose **GH-ESD** (Grounded Hypothesis-Driven Error Slice Discovery). As illustrated in Fig. 2, GH-ESD adopts a grounded hypothesis generate-and-verify paradigm. It integrates top-down semantic priors from a Large Language Model (LLM) with bottom-up region-level visual descriptions to construct a grounded hypothesis space of candidate failure patterns. These hypotheses are then verified using a Vision-Language Model (VLM) to ground them to specific visual regions. Crucially, to mitigate the hallucination risks of VLMs, we introduce a statistical hypothesis verification module that tests whether the model error probability increases monotonically with hypothesis intensity, thereby ensuring reliable and defensible slice identification.

Progress in this area is further limited by the absence of principled benchmarks. Existing slice discovery benchmarks focus primarily on image-level classification and artificial spurious correlations (e.g., Waterbirds [25], NICO++ [30]), which do not capture the complexity of real-world instance-level failures.

To enable standardized evaluation, we present **GESD** (Grounded Error Slice Dataset). GESD comprises expert-annotated, spatially grounded error slices

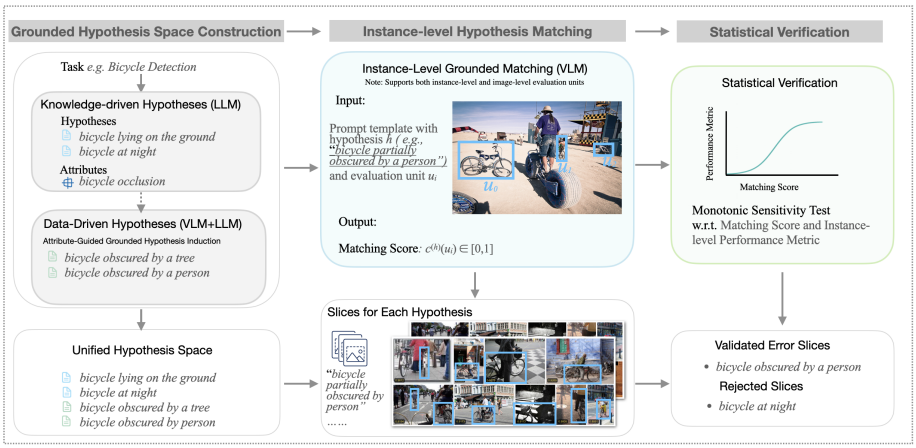


Fig. 2: Overview of GH-ESD. The framework constructs grounded hypotheses and verifies them through instance-level grounding and statistical trend analysis.

derived from real-world failures of multiple object detectors and a segmentation model, establishing a ground truth for instance-level error slice discovery.

Our main contributions are:

1. We propose **GH-ESD**, a grounded hypothesis generate-and-verify framework to robustly uncover systematic failure patterns. GH-ESD reframes instance-level error slice discovery as the construction of a grounded hypothesis space followed by statistical verification, enabling effective detection of spatially localized and contextual relational errors.
2. We establish **GESD**, which is the first benchmark dedicated to instance-level error slice discovery to our knowledge. It features expert-annotated ground-truth slices, representing spatially localized and contextual relational visual difficulties.
3. Extensive experiments demonstrate that GH-ESD consistently outperforms baselines, improving Precision@10 by 0.10 (0.73 vs. 0.63) on the GESD benchmark for detection tasks, while also supporting segmentation scenarios. We further validate the utility of discovered slices through model repair experiments, showing significant performance gains in object detection.

2 Related Work

Slice Discovery as Representation or Attribute Partitioning. Most existing slice discovery methods implicitly treat slices as partitions in representation space or combinations of predefined attributes. Early approaches identify failure modes by clustering embeddings or fitting mixture models in feature space [6, 8, 14, 27, 29]. For instance, Domino [8] and FACTS [29] discover systematic errors by modeling structured subgroups in embedding space. While

effective for image-level classification, these methods often lack interpretability, as slices correspond to regions in latent space rather than human-understandable concepts. To improve interpretability, subsequent work adopts a "tag-then-slice" paradigm, first generating semantic attributes and then searching for attribute combinations associated with poor performance [4, 5, 11, 12, 15, 21, 23]. Although more interpretable, this formulation assumes that failure patterns can be decomposed into independent atomic tags. Such attribute enumeration becomes inadequate for capturing compositional or relational visual patterns and suffers from combinatorial explosion in complex scenarios.

Instance-Level Error Slice Discovery. Extending slice discovery to object detection and segmentation requires reasoning over individual instances and their spatial relationships. VISLIX [28] adopts a cluster-then-explain strategy, assisting domain experts through UI-driven ROI embedding exploration and summarizing clusters using LLMs. HiBug2 [5] generates descriptive tags for each object instance using VLMs and searches for tag combinations correlated with errors. While these methods move toward instance-level clustering or bag-of-tags representations, they still rely on image-level error computation for slice evaluation, limiting their ability to perform fine-grained instance-level analysis. In contrast, our work formulates instance-level slice discovery as grounded hypothesis space construction with statistical verification, explicitly modeling grounded patterns and probing their impact on model performance.

Benchmarks for Slice Discovery and Robustness. Existing slice discovery benchmarks primarily target image-level classification and simplified spurious correlations, such as Waterbirds [25], CelebA [20], and NICO++ [30]. While useful for studying bias and subgroup robustness, these datasets do not capture the relational and spatial complexity of failures in detection and segmentation tasks. OOD benchmarks for complex tasks, including COCO-O [22] and Fishyscapes [2], focus on distribution shift and anomaly detection rather than identifying interpretable, systematic failure slices. Consequently, they do not provide the grounded, slice-level annotations necessary for evaluating instance-level slice discovery methods. To address this gap, we introduce GESD to provide a principled evaluation testbed for instance-level slice discovery.

3 Grounded Hypothesis-Driven Error Slice Discovery

As illustrated in Fig. 2, we propose GH-ESD, a grounded hypothesis-driven framework that discovers instance-level error slices through hypothesis construction, matching, and statistical verification. We begin by formalizing the error slice discovery problem.

3.1 Problem Formulation

Let $\mathcal{U} = \{u_i\}_{i=1}^{N}$ denote a set of evaluation units for an instance-level vision task. Each evaluation unit corresponds to an object instance predicted correctly, or an error-associated region identified through prediction–ground-truth comparison,

including (i) a ground-truth instance corresponding to a false negative (FN), (ii) a predicted region corresponding to a false positive (FP), (iii) a prediction–ground-truth discrepancy region capturing localization or segmentation errors, or (iv) the whole image when global failure patterns are evaluated. This unified definition allows GH-ESD to handle FNs, FPs, localization mismatches and also global errors within a single framework. Both correct and erroneous regions are evaluated to compute the performance metric for each slice.

Let f be a model to be analyzed and $\ell(u_i)$ denote the performance metric (e.g., error rate, $1 - \text{IoU}$) defined on evaluation unit u_i.

Traditional slice discovery methods define a candidate subgroup via a binary function $g(u_i) \in \{0, 1\}$, forming $\mathcal{S}_g = \{u_i \mid g(u_i) = 1\}$. A subgroup is considered an error slice if $\mathbb{E}[\ell(u_i) \mid u_i \in \mathcal{S}_g] - \mathbb{E}[\ell(u_i)] > \tau$, for a predefined threshold τ. Such hard partitioning relies on discrete membership decisions and is sensitive to noisy slices and fluctuations in error rates within the data.

We instead associate each hypothesis $h \in \mathcal{H}$ with a continuous matching score $c^{(h)}(u_i) \in [0, 1]$, quantifying the degree to which evaluation unit u_i satisfies hypothesis h. Slice discovery is therefore reformulated as analyzing how model performance varies with respect to hypothesis intensity (quantified by the matching score): $\ell(u_i)$ conditioned on $c^{(h)}(u_i)$. A systematic error slice corresponds to a hypothesis for which model error increases consistently as the hypothesis is more strongly satisfied.

3.2 Grounded Hypothesis Space Construction

The goal of hypothesis construction is to define a hypothesis space $\mathcal{H}$ that systematically characterizes potential failure patterns of the model. A hypothesis $h \in \mathcal{H}$ is defined as a semantically interpretable visual condition that can be evaluated with respect to an evaluation unit. The hypothesis space covers both intrinsic task difficulties and dataset-specific distributional artifacts inspired by the categorization of errors in HiBug2 [5].

Knowledge-Driven Hypotheses: Modeling Intrinsic Task Difficulty. To capture inherent challenges of a task, we adopt a top-down hypothesis generation strategy guided by LLMs (prompt in supplementary Fig. S1). Given a concise task description, the LLM leverages world knowledge to enumerate plausible failure scenarios. These hypotheses typically correspond to known sources of visual task difficulty, such as appearance ambiguity, scale variation, object interaction, and challenging environmental conditions. Importantly, these hypotheses are not tags but structured semantic conditions that can later be grounded and verified.

Data-Driven Hypotheses: Discovering Distributional Patterns. While knowledge-driven hypotheses capture general task difficulty, real-world datasets often exhibit distributional biases and rare configurations that are not fully anticipated by prior knowledge. To discover such patterns, we adopt a bottom-up grounded abstraction process, with prompt templates provided in supplementary Figs. S2, S3, and S4. *(a) Grounded Attribute-guided Caption Extraction.* For a sampled subset of images, VLM generates object- or region-grounded captions based on the attributes from LLM world knowledge to avoid divergent descriptions.

This attribute-guided design mitigates noise caused by unconstrained, overly diverse captions. Unlike image-level descriptions in TCIC [17], these captions explicitly associate semantic attributes with specific visual regions, preserving spatial grounding. *(b) Attribute and Value Inference.* An LLM analyzes the grounded captions to infer recurring semantic attributes and their candidate values (e.g., occlusion: by person, by car). This step identifies semantic dimensions present in the dataset. *(c) Hypothesis Synthesis.* The inferred attribute–value combinations are reformulated into well-formed natural-language hypotheses. Each resulting hypothesis specifies a concrete, evaluable visual condition that can be directly used as input to the matching stage.

This process transforms distributional regularities into interpretable and testable hypotheses. Knowledge-driven hypotheses provide broad coverage of intrinsic task difficulty. In contrast, data-driven hypotheses reveal dataset-specific distributional characteristics, which may include rare or long-tail patterns that are amplified in the target data distribution due to empirical frequency and contextual biases. For the task of bicycle detection, GH-ESD may generate knowledge-driven hypotheses such as "bicycle lying on the ground" or "bicycle at night", capturing intrinsic difficulty, and data-driven hypotheses such as "dataset-specific contextual configurations co-occurring with bicycles", reflecting dataset-induced regularities. These hypotheses are expressed in structured natural language, enabling direct interaction with vision-language models for grounded verification.

3.3 Hypothesis Matching

Given a hypothesis $h \in \mathcal{H}$ and evaluation unit u, we compute a continuous matching score $c^{(h)}(u) \in [0, 1]$, estimating the probability that the visual evidence associated with u satisfies h.

A pretrained VLM capable of grounding reasoning (e.g., Qwen2.5-VL [1]) is prompted with the hypothesis together with grounding signals (e.g., bounding boxes or point coordinates), and then produces token-level logits over its vocabulary. We extract logits for affirmative and negative tokens (e.g., 'yes'/'no') to compute a continuous score. Prompt templates are provided in supplementary Figs. S5 and S6.

Given prompt $\mathcal{P}(h, u)$, let $z(t \mid \mathcal{P}(h, u))$ denote the logit corresponding to token t. We compute the matching score as

$$c^{(h)}(u) = \frac{\exp\left(z(\text{yes} \mid \mathcal{P}(h, u))\right)}{\exp\left(z(\text{yes} \mid \mathcal{P}(h, u))\right) + \exp\left(z(\text{no} \mid \mathcal{P}(h, u))\right)}.$$

This probabilistic formulation converts the VLM's discriminative response into a normalized continuous score in $[0, 1]$.

3.4 Statistical Hypothesis Verification

Existing error slice discovery approaches often identify slices by thresholding empirical error rates within predefined subgroups. However, error rates can vary

substantially across patterns, making fixed thresholds difficult to choose and often unreliable. Moreover, a high subgroup error rate does not necessarily indicate a systematic failure mode. Such patterns may arise from noisy grouping, weak grounding, or hallucinated hypothesis matches. Bayesian correction [29] relies on explicitly modeled uncertainty over a predefined or stable tag space, which is incompatible with our open-ended hypothesis generation framework. We instead formulate hypothesis verification as a monotonic performance sensitivity test. Given a hypothesis h and its matching scores $\{c^{(h)}(u_i)\}_{i=1}^{N}$ over evaluation units $\{u_i\}_{i=1}^{N}$, we examine whether model error probability increases monotonically with hypothesis intensity.

The matching score $c^{(h)}(u)$ reflects the degree to which evaluation unit u satisfies hypothesis h. If h genuinely characterizes a failure mode of model f, units that more strongly satisfy h should exhibit higher error likelihood. Formally, we test whether the model performance metric shows a positive monotonic trend with respect to $c^{(h)}(u)$.

Low-confidence matches include irrelevant units due to matching intensity or VLM hallucination. Therefore, we restrict analysis to the relatively high-confidence region $\mathcal{U}_\gamma = \{u_i \mid c^{(h)}(u_i) > \gamma\}$, where γ is an easy-to-tune empirical threshold established via confidence calibration, varying across different VLMs.

Within $\mathcal{U}_\gamma$, evaluation units are sorted in descending order of $c^{(h)}(u_i)$. We apply a sliding window over the ordered units. Windows containing fewer than a minimum number of units (e.g., < 5) are discarded to ensure stability.

For each valid window w, we compute a local linear regression between matching score and performance metric:

$$\text{slope}_w = \text{Slope}\left(c^{(h)}(u), \ell(u)\right)_{u \in w}.$$

The overall trend statistic for hypothesis h is defined as $\text{Trend}(h) = \max_{w \in \mathcal{W}} \text{slope}_w$, where $\mathcal{W}$ denotes the set of valid windows. A hypothesis is retained as a systematic error slice if $\text{Trend}(h) > \tau_{\text{trend}}$, indicating a consistent monotonic increase in error probability with hypothesis intensity. Unless otherwise specified, we fix $\gamma = 0.5$ and $\tau_{\text{trend}} = 0.2$. As shown in Sec. 5.2, the method remains stable across a broad range of parameter settings.

The max sliding-window slope is lightweight and interpretable, and it detects failure patterns that manifest only when the hypothesis is strongly satisfied and mitigates spurious matches since hallucinated or weakly grounded units do not produce a stable positive slope. The underlying monotonic sensitivity principle can also be implemented using other monotonic association measures. We additionally instantiate a multi-scale Spearman rank-correlation test, retaining a hypothesis only when the association between matching score and error is both strong and statistically significant ($\rho \geq 0.5$, $p < 0.05$). Both the slope- and correlation-based tests are valid realizations of the monotonic strategy; we compare them quantitatively in Sec. 5.2.

4 GESD (Grounded Error Slice Dataset)

To address the limitations of existing benchmarks, we constructed GESD, a new benchmark dataset designed to capture realistic instance-level error slices. The GESD dataset is built upon a combination of over 12K images from three widely used public validation sets: COCO (5K images) [19], KITTI (3,769 images) [10], and a public face detection set (3,347 images) [7]. We followed a four-step pipeline to construct error slices and ensure the quality:

1. **Aggregated Model Errors Collection:** We ran four representative object detectors (YOLO [26], Faster R-CNN [24], RetinaNet [18] and DETR [3]) and a segmentation model (Mask R-CNN [13]) on the combined validation images. Instead of relying on a single model, we aggregated errors from multiple models to ensure a diverse and abundant collection of failure patterns. Specifically, we collected all false positive and false negative predictions from these models, forming a pool of candidate error instances.
2. **Expert-Driven Hypotheses Definition:** Based on the aggregated error pool, three senior computer vision researchers (each with over 4 years of experience) analyzed the failure patterns. They collaboratively defined a comprehensive and representative set of error slices by formulating hypotheses. Through iterative cross-validation, they identified 21 distinct hypotheses for detection and 21 for segmentation (total 42), categorizing them into semantic confusion, contextual interference, and intrinsic visual difficulties. This expert-driven process ensures that the defined slices correspond to meaningful and systematic failure modes rather than random noise.
3. **Hypotheses Matching by Human Annotation:** This step involved rigorous human annotation to match error instances to the defined hypotheses. A team of annotators, investing a total of 1500 person-hours, reviewed every relevant instance in the dataset. For each defined hypothesis, annotators (a) verified that the instance strictly met the slice definition and exhibited the correct failure type, and (b) corrected its ground-truth bounding box and/or segmentation mask to ensure geometric accuracy.
4. **Dataset Refinement:** To create a clean ground truth for evaluation, we performed a final refinement step. For all instances not belonging to any defined error slice, model predictions were overwritten by ground truth. This creates a synthetic "perfect" annotation for non-error parts and ensures evaluation focuses purely on predefined slices rather than open-world unknown errors.

The error slices in GESD contain location and category errors, including both false negatives (FN) and false positives (FP). We provide error regions with bounding boxes or segmentation masks, along with hypotheses or slice descriptions. Detailed slice statistics and error rates are provided in supplementary Sec. S2, and the full hypothesis list is available in supplementary Tabs. S3 and S4.

5 Experiments

We evaluate GH-ESD from four perspectives: (1) effectiveness on instance-level slice discovery, (2) ablation study of major components, (3) generalization to classical image-level bias benchmarks, and (4) practical utility via model repair.

We compare GH-ESD against state-of-the-art slice discovery methods, including: (1) embedding-based methods: FD (Feature Discovery) [14], Domino [8], and FACTS [29]; (2) interpretability-based methods: LADDER [11], ViG-FACTS [23], B2T (Bias-to-Text) [15], and ViG-B2T [23]. These approaches were originally designed for image-level bias analysis and do not explicitly or intrinsically support instance-level error slice discovery. To evaluate the proposed instance-level framework, we adapt FACTS [29] by converting bounding box predictions to image-level labels—if any bounding box in an image has an error, the entire image is marked as erroneous. HiBug [4] and HiBug2 [5] aggregate instance errors to image-level and can be adapted to extract tags for objects. We therefore compare the adapted FACTS*, HiBug* and HiBug2 on GESD. HiBug and HiBug2 originally used GPT-3.5-Turbo and GPT-4o. We reproduced them with GPT-5.2 to avoid underestimating these baselines, while preserving their original algorithms.

Our GH-ESD implementation employs Gemini-2.5-Pro as LLM and Qwen2.5-VL-7B as VLM, selected for a trade-off between performance and efficiency (the influence of selecting different models is evaluated in supplementary Sec. S3.1). Prompts of GH-ESD are provided in supplementary Sec. S1.1.

Error slices can be described at different levels of granularity, so predicted and ground-truth slices may split or merge and do not correspond one-to-one. Consequently, recall becomes sensitive to the slice matching criteria, and prior work [8, 29] primarily evaluates slice discovery using Precision@k. This metric measures the precision of the top-k retrieved samples for each ground-truth error slice. Formally, for a ground truth slice S_{gt} and predicted slice S_{pred}, Precision@k is defined as:

$$\text{Precision@k} = \frac{|S_{gt} \cap S_{pred}^{(k)}|}{k} \tag{1}$$

where $S_{pred}^{(k)}$ represents the top-k samples in the slice ranked by hypothesis matching scores. Following prior work, we set $k = 10$ by default, or the actual size if the slice has fewer than 10 samples.

The **Precision@k** evaluates accuracy of instances belonging to slices without considering hypothesis effectiveness; thus, the hypothesis precision and recall will be provided in Sec. 5.2.

5.1 Results on GESD Instance-Level Benchmark

We first evaluate GH-ESD on GESD and compare against baselines: FACTS* (adapted via image-level aggregation), HiBug* (extended to instance-level tagging) and HiBug2. For the tag-based baselines, we reproduce their tag extraction

Table 1: Precision@k results on GESD dataset.

Metric	FACTS*	HiBug*	HiBug2 (GPT)	HiBug2 (Gemini)	GH-ESD
P@5	0.35	0.34	0.73	0.77	**0.78**
P@10	0.28	0.31	0.63	0.63	**0.73**
P@20	0.20	0.26	0.53	0.52	**0.66**

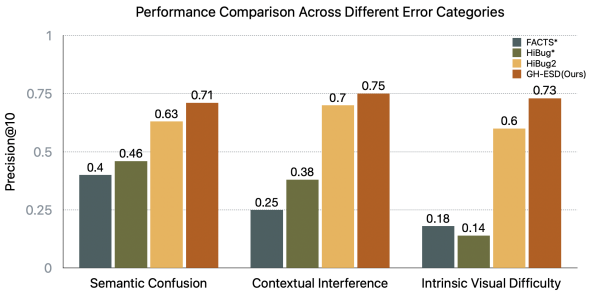


Fig. 3: Performance comparison across different error categories.

procedures. HiBug* tags are interactively supplemented with instance-level attributes derived from detected objects (supplementary Sec.S3.2), while HiBug2 attributes are extracted using the released implementation.

Table 1 reports the Precision@k results on the GESD detection-based dataset. On Precision@10, the metric most commonly adopted in prior slice-discovery work [8,29], GH-ESD achieves 0.73, strongly outperforming HiBug2 (0.63), HiBug* (0.31) and FACTS* (0.28)—an absolute improvement of 0.10 over the strongest baseline—and it leads across all metrics, with the largest gains at P@10 and P@20. For a fair comparison, we further evaluated HiBug2 with the same Gemini-2.5-Pro backend used by GH-ESD; the results indicate that the performance gains stem from our instance-level framework rather than the choice of LLM. These improvements are attributed to the accurately described hypotheses, grounding capability, and statistical hypothesis verification of our instance-level framework.

Broken down by failure mode, GH-ESD also demonstrates superior robustness: it achieves a P@10 of 0.64 for false positives (FPs) and 0.79 for false negatives (FNs), compared to HiBug2's 0.51 and 0.72, respectively. This confirms that attribute-based tagging struggles with relational context, especially for context-heavy errors, whereas our continuous VLM reasoning over coherent natural-language hypotheses handles both FPs and FNs effectively.

To further analyze performance across error categories, Fig. 3 presents category-wise results under our slice taxonomy: (1) Semantic Confusion, covering errors arising from limitations in the model's conceptual understanding; (2) Contextual Interference, encompassing errors caused by an object's surroundings or relationships; and (3) Intrinsic Visual Difficulty, referring to failures caused by

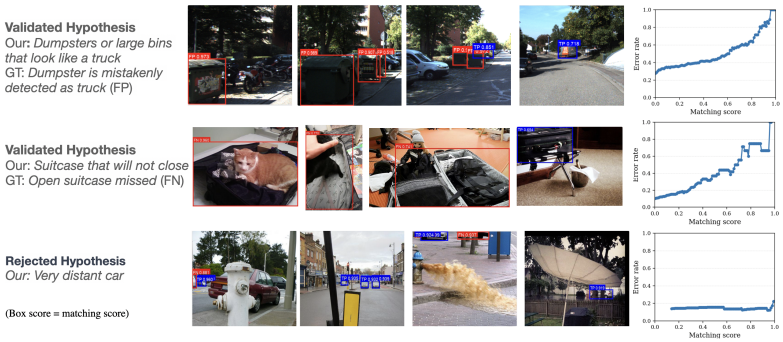


Fig. 4: Examples of candidate hypotheses and their verification outcomes. The right column shows the statistical trend between matching score and error rate for each hypothesis.

the inherent visual properties of the object instance itself. GH-ESD consistently outperforms baselines across all categories.

Figure 4 shows several candidate hypotheses and their verification outcomes. Hypotheses such as dumpsters that visually resemble trucks and suitcases that will not close correspond to clear systematic error patterns that consistently lead to FPs or FNs. In contrast, the hypothesis of very distant cars exhibits inconsistent error trends and therefore does not form a reliable slice. Other hypotheses, such as cars occluded severely by trees, lack sufficient supporting samples, making their verification inconclusive. These examples demonstrate that our framework not only discovers meaningful error slices but also filters out spurious or statistically unsupported hypotheses.

Segmentation-based slice discovery results are reported in supplementary Tab. S4, demonstrating that GH-ESD generalizes across both detection and segmentation tasks. The bounding box of the erroneous pixel region is used as the evaluation unit, and these results serve as baseline references for future studies.

5.2 Ablation Study of GH-ESD

We now analyze the contribution of each major component: (1) grounded hypotheses, (2) instance-level hypothesis matching, and (3) statistical hypothesis verification.

Grounded Hypotheses vs. Atomic Tags We compare the proposed language-based hypotheses with attribute-based tags (HiBug2) in terms of their semantic alignment with the ground-truth failure slices, using results from Sec. 5.1. Specifically, an LLM-based (Gemini-2.5-Pro) evaluator determines whether a discovered slice is semantically related to any ground-truth slice (prompt in supplementary Sec. S1.3). A slice is considered correct if the evaluator judges it as relevant. *Recall* measures the proportion of ground-truth slices that are successfully retrieved,

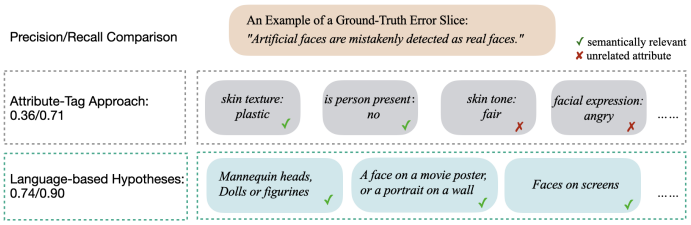


Fig. 5: Comparison between tags and our hypotheses on semantic relevance with ground truth error slices.

while *precision* measures the proportion of discovered slices that are semantically aligned with the ground truth.

The language-based hypotheses (102 slices) achieve *precision/recall* of 0.74/0.90 when evaluated for semantic relevance to ground-truth error slices. In contrast, the attribute-based tagging method HiBug2 (3,666 slices) generates a large number of attribute combinations (183/713/2,770 for single/dual/triple-attribute combinations), yielding *precision/recall* of 0.36/0.71 under a loose matching criterion. Figure 5 illustrates that language-based hypotheses capture the semantic structure of real-world failure patterns more coherently and compactly, while tag combinations tend to provide general semantic descriptions that lack specific failure contexts.

This performance gap reflects structural limitations of tag-based slices in the detection benchmark. Several failure modes involve interaction- or context-dependent patterns, such as "bicycle partially occluded by a person," and "a dumpster that looks like a truck." These conditions cannot be faithfully represented as simple conjunctions of atomic tags (e.g., object + is occluded), since atomic tags are semantically divergent and cannot readily capture relational directionality, embedded context, or visual similarity relations. Language-based hypotheses capture these structured patterns directly.

Furthermore, Knowledge-Driven (KD) and Data-Driven (DD) hypotheses exhibit strong complementarity; further analysis of the experiments in Tab. 1 reveals that KD-only and DD-only achieve P@10 scores of 0.62 and 0.21, respectively, yet their combination reaches 0.73 with a 14% pattern overlap.

Instance-Level Grounding vs. Image-Level Aggregation We evaluate the importance of spatial grounding by comparing instance-level conditioning with image-level aggregation on the fixed hypothesis "faces partially hidden by objects." Instance-level grounding achieves Precision@10 of 0.80, much higher than 0.60 under image-level analysis. Moreover, trend-based verification yields a stronger matching score-error trend statistic under instance-level conditioning than under image-level evaluation (1.32 vs. 0.84), confirming that instance-level conditioning mitigates dilution effects where correct predictions on unrelated instances obscure localized failures. The LLM and VLM are kept fixed in this

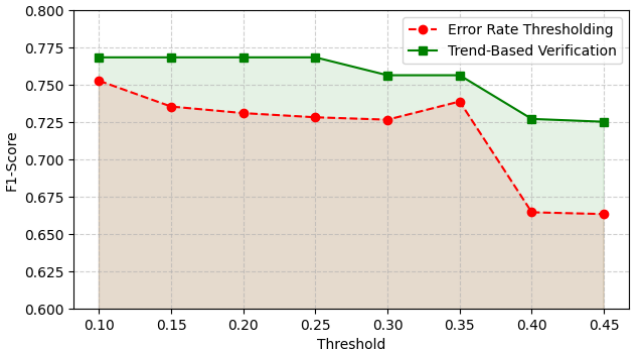


Fig. 6: Comparison between error rate thresholding and statistical hypothesis verification.

experiment, suggesting that the performance gains primarily stem from the grounding mechanism.

Statistical Hypothesis Verification vs. Error-Rate Thresholding We compare our monotonic trend-based hypothesis verification with traditional error-rate thresholding [5, 11] using results from Sec. 5.1. For all discovered error slices, we compute the recall of ground-truth consistent hypotheses, the precision of validated error slices with respect to the ground truth, and report the corresponding F1 score. As shown in Fig. 6, trend-based verification achieves an average 3.7% improvement in F1 score and demonstrates greater robustness to threshold variation. We further evaluate the multi-scale Spearman variant defined in Sec. 3.4, which achieves an F1 score of 0.78 compared to 0.77 using the original slope, and remains stable under Benjamini–Hochberg FDR correction for multiple-hypothesis control. These suggest that the effectiveness of our approach arises from the monotonic sensitivity principle itself.

5.3 Generalization to Image-Level Bias Benchmarks

Although GH-ESD is designed for instance-level reasoning, we evaluate our method on three bias discovery benchmarks: (1) Waterbirds [25], which contains two error-prone slices used for bias evaluation: waterbirds on land and landbirds on water; (2) CelebA [20], where, following [29], we focus on the blonde hair classification task that exhibits a spurious correlation between gender and hair color; and (3) NICO++ [30], where we simulate controlled spurious-correlation settings following FACTS [29] and evaluate under three correlation strengths—75%, 90%, and 95%—representing increasing levels of spurious association between concepts and contexts. Results are summarized in Tab. 2. GH-ESD achieves competitive or state-of-the-art Precision@10 across datasets, and performs particularly well under complex spurious correlation settings on NICO++. Implementation details are provided in supplementary Sec. S3.4.

Table 2: Precision@10 results on bias discovery datasets. Best results are shown in **bold**, second-best are underlined.

Method	Waterbirds	CelebA	NICO++(75/90/95)
FD	0.90	0.70	0.19/0.19/0.19
Domino	**1.00**	0.90	0.24/0.25/0.27
FACTS	**1.00**	0.90	0.56/0.60/0.62
LADDER	0.86	0.85	0.32/0.52/0.54
ViG-FACTS	**1.00**	**1.00**	0.60/ **0.67**/0.65
B2T	0.92	0.64	-
ViG-B2T	0.97	0.70	-
GH-ESD (Ours)	**1.00**	**1.00**	**0.64**/0.66/**0.69**

Table 3: Model repair performance on bicycle detection. Results are averaged over 3 repetitions.

Method	mAP-bicycle	mAR-bicycle
No model repair	27.20	39.33
Baseline model repair	28.58 ± 0.016	41.14 ± 0.031
GroupDRO + **GH-ESD**	**33.59** ± 0.028	**47.09** ± 0.230

5.4 Model Repair and Practical Impact

Repair on Object Detection Benchmarks. We evaluate whether GH-ESD can facilitate targeted model repair for object detection (e.g., bicycles). After verifying the error slice *"bicycle partially occluded by a person"* on GESD, we transfer the discovered hypothesis to the COCO training set and compute instance-level slice matching scores using GH-ESD. As a baseline, we fine-tune a pre-trained Faster R-CNN on all COCO training images containing bicycle instances. For slice-aware repair, we adapt GroupDRO [25] to emphasize failure-prone samples identified by GH-ESD. Specifically, images with high slice confidence are upweighted during fine-tuning, and reweighting is applied to the detection head losses. Details are in supplementary Sec. S3.5.

As shown in Tab. 3, slice-aware repair significantly improves bicycle-class mAP and mAR over the baseline. These results demonstrate that grounded, instance-level slice discovery enables more effective model adaptation.

Repair on Classification Benchmarks. We evaluate slice actionability on CelebA and Waterbirds following [11]. Due to fundamental differences in slice definition strategies, we employ two repair protocols. For LADDER, which produces global slice definitions, we adopt its native strategy of training a classifier per slice and ensembling their predictions. For FACTS, Domino, and GH-ESD, whose slices are defined over data subsets, we instead apply a unified DFR-based repair framework [16]. Specifically, we compute a per-sample failure score as the

Table 4: Worst Group Accuracy (WGA) results. Best results are shown in **bold**, second-best are underlined. All results are averaged over 10 independent runs to account for the stochasticity of SGD during fine-tuning.

Method	Waterbirds	CelebA
LADDER	0.896 ± 0.014	0.878 ± 0.015
Domino	0.898 ± 0.022	0.827 ± 0.002
FACTS	0.889 ± 0.028	0.600 ± 0.011
GH-ESD (Ours)	**0.904** ± 0.017	**0.882** ± 0.004

maximum similarity over discovered slices, and fine-tune the classifier head on high- and low-scoring subsets. Details are in supplementary Sec. S3.5.

As shown in Tab. 4, GH-ESD achieves comparable or better performance than existing slice discovery baselines, indicating that the discovered slices effectively mitigate errors induced by spurious correlations in image classification.

6 Conclusion

We presented **GH-ESD**, a spatially grounded, hypothesis-driven framework for instance-level error slice discovery, and introduced **GESD**, the expert-annotated benchmark enabling evaluation of instance-level slice discovery.

Discussion and Future Work. Error slice discovery is inherently constrained by the coverage and bias of the evaluation dataset. When certain failure modes are underrepresented, purely statistical analysis may be insufficient to reveal them. Moreover, the hypothesis space in real-world deployments is fundamentally open-ended, and automatically generated hypotheses may not guarantee completeness. These observations suggest that scalable error slice discovery will likely require tighter integration between automated discovery and human reasoning. Promising directions include dynamically incorporating domain-specific knowledge sources, developing adaptive hypothesis expansion mechanisms, and exploring more principled human-in-the-loop strategies for interactive debugging. Our interactive visualization tool, presented in supplementary Sec. S4, provides an initial step toward this direction.

References

1. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-VL technical report. arXiv preprint arXiv:2502.13923 (2025)
2. Blum, H., Sarlin, P.E., Nieto, J., Siegwart, R., Cadena, C.: The fishyscapes benchmark: Measuring blind spots in semantic segmentation. IJCV **129**(12), 3119–3135 (2021)

3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV. pp. 213–229 (2020)
4. Chen, M., Li, Y., Xu, Q.: HiBug: On human-interpretable model debug. In: NeurIPS. vol. 36, pp. 4753–4766 (2023)
5. Chen, M., Zhao, C., Xu, Q.: HiBug2: Efficient and interpretable error slice discovery for comprehensive model debugging. In: ICLR (2025)
6. d'Eon, G., d'Eon, J., Wright, J.R., Leyton-Brown, K.: The spotlight: A general method for discovering systematic errors in deep learning models. In: Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT). pp. 1962–1981 (2022)
7. Elmenshawii, F.: Face detection dataset. Kaggle dataset. `https://www.kaggle.com/datasets/fareselmenshawii/face-detection-dataset/versions/2` (2025), accessed: Aug. 22, 2025
8. Eyuboglu, S., Varma, M., Saab, K., Delbrouck, J.B., Lee-Messer, C., Dunnmon, J., Zou, J., Ré, C.: Domino: Discovering systematic errors with cross-modal embeddings. ICLR (2022)
9. Gannamaneni, S.S., Rao, R.P., Mock, M., Akila, M., Wrobel, S.: Detecting systematic weaknesses in vision models along predefined human-understandable dimensions. Transactions on Machine Learning Research (TMLR) (2025)
10. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: CVPR. pp. 3354–3361 (2012)
11. Ghosh, S., Syed, R., Wang, C., Choudhary, V., Li, B., Poynton, C.B., Visweswaran, S., Batmanghelich, K.: LADDER: Language-driven slice discovery and error rectification in vision classifiers. In: Findings of the Association for Computational Linguistics: ACL. pp. 22935–22970 (2025)
12. Guimard, Q., D'Incà, M., Mancini, M., Ricci, E.: Classifier-to-Bias: Toward unsupervised automatic bias detection for visual classifiers. In: CVPR. pp. 15151–15161 (2025)
13. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: ICCV. pp. 2961–2969 (2017)
14. Jain, S., Lawrence, H., Moitra, A., Madry, A.: Distilling model failures as directions in latent space. In: ICLR (2023)
15. Kim, Y., Mo, S., Kim, M., Lee, K., Lee, J., Shin, J.: Discovering and mitigating visual biases through keyword explanation (Bias-to-Text). In: CVPR (2024)
16. Kirichenko, P., Izmailov, P., Wilson, A.G.: Last layer re-training is sufficient for robustness to spurious correlations. In: ICLR (2023)
17. Kwon, S., Park, J., Kim, M., Cho, J., Ryu, E.K., Lee, K.: Image clustering conditioned on text criteria. In: ICLR (2024)
18. Lin, T., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. IEEE TPAMI **42**(2), 318–327 (2020)
19. Lin, T., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollár, P.: Microsoft COCO: Common objects in context. In: ECCV. pp. 740–755 (2014)
20. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: ICCV. pp. 3730–3738 (2015)
21. Luo, Y., An, R., Zou, B., Tang, Y., Liu, J., Zhang, S.: LLM as dataset analyst: Subpopulation structure discovery with large language model. In: ECCV. pp. 235–252. Springer (2024)
22. Mao, X., Chen, Y., Zhu, Y., Chen, D., Su, H., Zhang, R., Xue, H.: COCO-O: A benchmark for object detectors under natural distribution shifts. In: ICCV (2023)

23. Marani, B., Hanini, M., Malayarukil, N., Christodoulidis, S., Vakalopoulou, M., Ferrante, E.: ViG-Bias: Visually grounded bias discovery and mitigation. In: ECCV. pp. 414–429. Lecture Notes in Computer Science, Springer (2024)
24. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE TPAMI **39**(6), 1137–1149 (2017)
25. Sagawa, S., Koh, P.W., Hashimoto, T.B., Liang, P.: Distributionally robust neural networks. In: ICLR (2020)
26. Wang, C.Y., Bochkovskiy, A., Liao, H.Y.M.: YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: CVPR. pp. 7464–7475 (2023)
27. Wang, F., Adebayo, J., Tan, S., Garcia-Olano, D., Kokhlikyan, N.: Error discovery by clustering influence embeddings. In: NeurIPS. vol. 36, pp. 41765–41777 (2023)
28. Yan, X., Xuan, X., Ono, J.P., Guo, J., Mohanty, V., Kumar, S.A., Gou, L., Wang, B., Ren, L.: VISLIX: An XAI framework for validating vision models with slice discovery and analysis. In: Computer Graphics Forum (CGF). p. e70125 (2025)
29. Yenamandra, S., Ramesh, P., Prabhu, V., Hoffman, J.: First amplify correlations and then slice to discover bias (FACTS). In: ICCV (2023)
30. Zhang, X., He, Y., Xu, R., Yu, H., Shen, Z., Cui, P.: Nico++: Towards better benchmarking for domain generalization. In: CVPR. pp. 16036–16047 (2023)
31. Zhou, E., Su, Q., Chi, C., Zhang, Z., Wang, Z., Huang, T., Sheng, L., Wang, H.: Code-as-monitor: Constraint-aware visual programming for reactive and proactive robotic failure detection. In: CVPR. pp. 6919–6929 (2025)