Component-Based Out-of-Distribution Detection

Wenrui Liu[1,2] **Hong Chang**[1,2*] **Ruibing Hou**[1] **Shiguang Shan**[1,2]
[1] Institute of Computing Technology, Chinese Academy of Sciences, China
[2] University of Chinese Academy of Sciences, China
wenrui.liu@vipl.ict.ac.cn {changhong,houruibing,sgshan}@ict.ac.cn

Abstract

Out-of-Distribution (OOD) detection requires sensitivity to subtle shifts without overreacting to natural In-Distribution (ID) diversity. However, from the viewpoint of *detection granularity*, global representation inevitably suppresses local OOD cues, while patch-based methods are unstable due to entangled spurious-correlation and noise. And neither of them is effective in detecting *compositional* OODs composed of valid ID components. Inspired by recognition-by-components theory, we present a post-hoc **Component-Based OOD Detection (CoOD)** framework that addresses the existing limitations by decomposing inputs into functional *components*. To instantiate CoOD, we derive **Component Shift Score (CSS)** to detect local appearance shifts, and **Compositional Consistency Score (CCS)** to identify cross-component compositional inconsistencies. Empirically, CoOD achieves consistent improvements on both coarse- and fine-grained OOD detection.

1 Introduction

Real-world computer vision systems must not only produce accurate predictions on in-distribution (ID) inputs, but also abstain on out-of-distribution (OOD) inputs to avoid risky decisions. Previous work on OOD detection has explored multiple complementary directions, such as designing scoring functions to reduce model overconfidence [1–3] and learning more discriminative representations to improve detection [4–6]. Orthogonal to these directions, *detection granularity* introduces another axis, determined by the spatial level at which evidence is aggregated. From this viewpoint, OOD detection approaches can be broadly categorized into global or local ones.

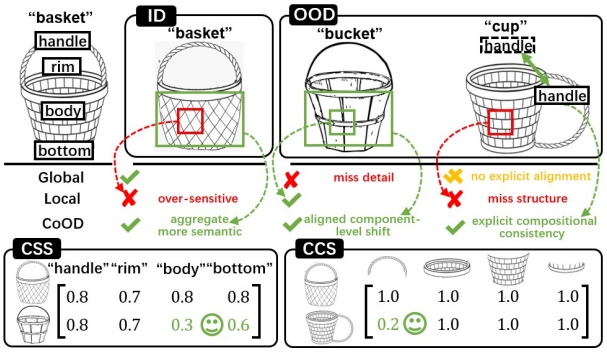

Figure 1: Comparison between CoOD and global/local methods. **Top:** From left to right are: training ID *"basket"*, test ID *"basket"*, fine-grained OOD *"bucket"*, and compositional OOD *"cup"*. **Middle:** Causes of success or failure. **Bottom:** CSS exposes fine-grained shift in components *"bucket's body"*, while the CCS detects a compositional misalignment in *"cup's handle"*.

Global approaches [1–3] aggregate features across the entire image into a holistic representation and perform well under large semantic shifts. However, by global pooling or features aggregation, they project inputs onto dominant class-semantic directions. Consequently, they smooth out, and thereby weaken, the localized deviations [7] that are crucial for ***detecting fine-grained OODs***. As shown in

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Figure 1, the global method overlooks the texture difference in the *"body"* of *"bucket"*, thus fails to detect it as OOD.

By contrast, *local (patch-based) approaches* [8, 9] capture fine-grained OOD by preserving spatially dense information [10]. However, they are inherently prone to the ***sensitivity-robustness dilemma***: fine-grained evidence increases sensitivity to subtle OODs, but simultaneously exposes the detector to patch noise, cross-patch contamination [11], and model bias [12], thereby undermining robustness. Furthermore, simple post-hoc strategies that select salient embeddings [13] or local patches [8, 9, 14] still suffer from semantic confusion and low signal-to-noise ratio, as illustrated in Figure 1.

Another challenge concerns ***detecting compositional OODs*** constructed from components similar to ID parts. Local schemes that evaluate tokens independently cannot, by design, determine whether the global context is inconsistent, as they lack an explicit model of inter-token compositional relations[2]. As shown in Figure 1, the OOD *"cup"* exhibits similar local presence with ID *"basket"*, thus can hardly be detected. Surprisingly, global methods also fall short for this case: uneven feature-learning [15] and limited receptive fields [16] cause the model to rely on a small subset of salient evidence, while inherent modeling biases [17, 18] further restrict its ability to encode compositional structure.

An ideal OOD detector should resolve the sensitivity-robustness dilemma, *i.e.* tolerating reasonable ID variation and mild noise, while rejecting semantic or structural OODs at *any* scale [19]. For this purpose, we introduce **Component-Based OOD Detection (CoOD)**, a component-centric post-hoc paradigm, motivated by Recognition-by-Components theory [20]. CoOD decomposes instances into **functional components** that serve as *the finest meaningful granularity* to identify local OODs while averaging out patch-level noise, *i.e.* finer-grained aggregation adds little signal but more noise, whereas coarser pooling suppresses local OOD cues. On this basis, component relationship further supplies global semantic and geometric context that complements component-level evidence.

Crucially, CoOD leverages two complementary evidence streams. **Component Shift Score (CSS)** aggregates tokens within each component to construct robust representations. By averaging intra-component variations, CSS suppresses patch-level noise and contextual leakage, preserving component semantics and amplifying sensitivity to fine-grained OODs. **Compositional Consistency Score (CCS)** measures semantic and geometric consistency across components by matching component structure to a compact ID coreset and estimating spatial alignment. Thus, large alignment residuals or low feature agreements indicate invalid composition. We further use a plug-and-play *non-component suppression* step to reduce background and cross-component interference. As Figure 1 shows, CSS captures the *"body"* shift (*w.r.t.* texture) between *"basket"* and *"bucket"*, while CCS flags a *"handle"* inconsistency that identifies *"cup"* as OOD. This combination yields *inspectable component-level* evidence and improves both coarse- and fine-grained OOD detection without retraining.

Our contributions are threefold:

- We propose a post-hoc, component-centric OOD detector, providing inspectable component-level evidence and reconciling fine-grained sensitivity with robustness to ID variations.
- We instantiate CoOD via a component-aware aggregation framework, **C**omponent **S**hift **S**core for fine-grained OODs and **C**ompositional **C**onsistency **S**core for compositional OODs.
- We theoretically analyze the component-level evidence in OOD detection and empirically demonstrate the effectiveness of CoOD, particularly for fine-grained and compositional OOD.

2 Related Works

Global OOD Detection. A major line of OOD research focuses on globally aggregated representations, most commonly classifier outputs or pooled embeddings. Representative detectors include maximum softmax probability [21], energy score [1, 22], density measure [23], recent logit refinements [3] and ensemble learning [24]. These methods perform well for large semantic shifts because global representation amplifies the dominant class signals.

With the advent of vision-language models and zero-shot setting [25], CLIP-based detectors have stronger ability in measuring similarity to textual representation [2], or expanded pseudo-OOD sets [26–28]. However, such detectors still fall short due to the holistic measure, which tends to suppress

[2]This failure pattern is pervasive in practice, as semantic shifts often accompany by structural misalignments.

localized appearance discrepancies. To address this limitation, CoOD preserves meaningful localized signals via component-guided aggregation, retaining sensitivity to fine-grained OODs.

Local OOD Detection. To recover details suppressed by global aggregation, some recent works incorporate spatial locality into OOD detection, *e.g.* computing dense (patch-level) similarity [8, 14, 29] or implicitly enhancing locality via shallow visual feature comparisons [13, 30]. Others use training-time regularization, *e.g.*, treating background patches as negative signals [31] for outlier exposure [32], prompt tuning [6, 9], or localized contrastive learning [14, 29].

Although these strategies increase local sensitivity, they either require additional training or exhibit limited stability, because patch representations remain entangled with spurious correlations [11], model priors [12], and incidental noise. Crucially, they cannot capture structural relationships among local patches. In contrast, CoOD is designed to suppress irrelevant noise and obtain robust component semantics, addressing both patch-level instability and the lack of compositional awareness.

Component-based Methods. From recognition-by-components theory [20], robust vision lies in decomposing objects into components and reasoning over their compositions. This insight has inspired modern deep models to explicitly capture intermediate visual components. For example, [33, 34] extract components to support interpretable evidence-based classification. [35] further develops stable part discovery by non-parametric clustering. Beyond individual parts, explicit compositional modeling can also improve robustness under occlusion and semantic variation [36, 37].

Although modern networks implicitly encode component cues, they are often entangled with global context and noise, limiting robustness under semantic variations and sensitivity to fine-grained and compositional OODs. Simply combining global and local scores is insufficient, as post-hoc mixing is merely a compromise, rather than disentangling nuisance factors that conflate local evidence with noise and global biases. To our knowledge, CoOD is the first OOD detector that decomposes inputs into components and provides principled criteria for both fine-grained and compositional OODs.

3 Methodology

This section starts by the problem setup and our theoretical intuition. We then describe how to construct CoOD by component-level representations, CSS and CCS.

3.1 Preliminaries

Let $\mathcal{X}$ and $\mathcal{Y}=\{1,2,\dots,C\}$ be the input space and semantic label space, with joint distribution $P_{\mathcal{X}\times\mathcal{Y}}$. During the test phase, an OOD detector observes only $\boldsymbol{x}\in\mathcal{X}$ and decides whether $\boldsymbol{x}$ comes from the ID marginal distribution $\boldsymbol{x}\sim P_{\mathcal{X}}$ or from OODs. This follows the conventional post-hoc semantic OOD detection setting [1, 2, 21].

Given visual encoder $f:\mathcal{X}\to\mathcal{Z}$, test sample $\boldsymbol{x}$ is encoded as $\boldsymbol{z}=f(\boldsymbol{x})$. We denote position representations collectively as $\boldsymbol{e}$ and the total number of spatial positions as N. For vision-language models with a text encoder g, the semantic space is defined by the encoded text $\boldsymbol{t}_y\in\{g(\text{“text}_y\text{”})\}_{y=1}^{C}$. Logit activation between visual $\boldsymbol{z}$ and textual $\boldsymbol{t}_y$ is given by cosine similarity $\langle\boldsymbol{z},\boldsymbol{t}_y\rangle=\frac{\boldsymbol{z}^\top\boldsymbol{t}_y}{\|\boldsymbol{z}\|\|\boldsymbol{t}_y\|}$ and the posterior is $P_\theta(y\,|\,\boldsymbol{x})=e^{\langle\boldsymbol{z},\boldsymbol{t}_y\rangle/\tau}/\sum_{y'\in\mathcal{Y}}e^{\langle\boldsymbol{z},\boldsymbol{t}_{y'}\rangle/\tau}$ with a temperature factor τ.

For CoOD, each y is associated with a component set $\mathcal{P}_y$. Following recognition-by-components [20], we define a component as a class-relevant, visually discriminative part that is spatially localizable to provide reliable ID evidence. For each $p\in\mathcal{P}_y$, the semantics are encoded as $\boldsymbol{t}_{yp}\in\{g(\text{“text}_{yp}\text{”})\}_{p=1}^{|\mathcal{P}_y|}$. With foreground mask $\boldsymbol{m}_y\in[0,1]^{H\times W}$ and component masks $\boldsymbol{m}_{yp}\in[0,1]^{H\times W}$, we compute the weighted average of two complementary evidences: $\mathrm{CSS}(\boldsymbol{x})+\alpha\mathrm{CCS}(\boldsymbol{x})$.

3.2 Theoretical Motivation

As illustrated by *"Foreground"* in Figure 3, discriminative models often concentrate gradient mass on a small set of foreground components. Such models may change their confidence sharply in the presence of ID variations, and miss fine-grained OOD shifts that occur beyond the focused regions. CoOD can address these issues from three aspects: (1) explicitly including component representations to mitigate the neglect of detection-relevant fine-grained regions (Section 3.2.1),

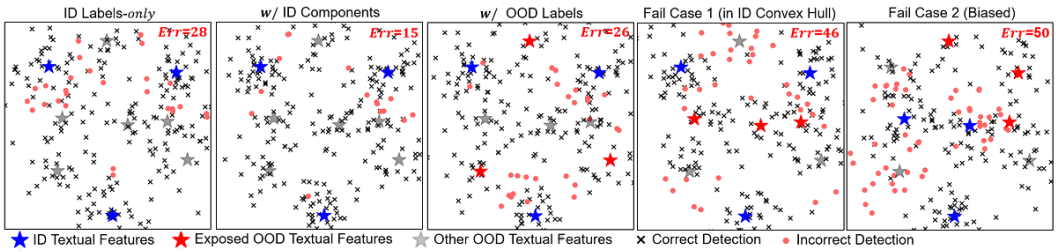

Figure 2: Comparison of sensitivity to text selection between CoOD and negative-label-based methods on a toy dataset. From left to right are: baseline (MCM), CoOD (components are drawn around the corresponding ID classes), ideal OOD label-based scenario (EOE), and two failure cases (EOE). Err (•) indicates the number of incorrect detection. Since CoOD has correct detection in the most of component perspectives, the aggregated score has lower variance and Err.

(2) reducing confusion from irrelevant context (Section 3.2.2), and (3) supplying compositional evidence of inter-component relationships [20]. We theoretically analyze the first two aspects under a simplified component-evidence model, while validating the third empirically.

3.2.1 Introducing component representations

Prior work [5] shows that auxiliary information can improve OOD detection. Following the widely used setup [27, 28, 30], we define an *existence score* for each component p as $s_{yp} = \langle \boldsymbol{z}_{yp}, \boldsymbol{t}_{yp} \rangle$ and model the component-level evidence as a Bernoulli random variable with parameter $\psi = P(s_{yp} > \gamma | \boldsymbol{x})$, which reflects the probability of being an ID component. The aggregated ID score $\mathcal{S} = \sum_p \mathbb{1}\{s_{yp} > \gamma\}$ can be modeled by a Binomial distribution $\mathcal{B}(|\mathcal{P}_y|, \psi)$, and approximated[3] by Normal distribution as $\mathcal{N}(|\mathcal{P}_y|\psi, |\mathcal{P}_y|\psi(1-\psi))$. The false positive rate at true positive rate λ is:

$$\text{FPR}_\lambda = \Phi\Big((\mu^{\text{out}} - \mu^{\text{in}})/\sigma^{\text{out}} + \Phi^{-1}(\lambda)\sigma^{\text{in}}/\sigma^{\text{out}}\Big), \tag{1}$$

where Φ is monotonic standard normal CDF and (μ, σ^2) parameterize the score distribution for ID and OOD samples. Differentiating FPR_λ *w.r.t.* $|\mathcal{P}_y|$ gives:

$$\partial\, \text{FPR}_\lambda / \partial\, |\mathcal{P}_y| \;\propto\; (\psi^{\text{out}} - \psi^{\text{in}}) / 2\sqrt{|\mathcal{P}_y| \psi^{\text{out}} (1 - \psi^{\text{out}})}. \tag{2}$$

Thus adding component evidences $|\mathcal{P}_y| \uparrow$ reduces FPR_λ provided $\psi^{\text{in}} > \psi^{\text{out}}$. Some previous methods alternatively leverage additional information from OOD labels [26, 27], but their effectiveness depends heavily on the feature-space structure between ID and OOD, and may fail in practice, *e.g.* OOD labels may fall inside the convex hull of IDs, producing opposing signals as shown in Figure 2.

In contrast, CoOD introduces components near ID textual representation to enhance ID proximity and OOD discriminability. Real OODs tend to be distant from most ID components, whereas IDs remain close to at least some. By aggregating component evidence, CoOD suppresses irrelevant noise (from single patch and regions beyond the targeted component) and prevents easy-to-learn [38] or dominant features from attenuating other component signals, thereby improving fine-grained OOD detection.

3.2.2 Reducing nuisance correlations

However, in practice, visual representations extracted from the *whole* image may entangle the target component with context contamination or other spurious cues (*e.g.*, backgrounds or spurious correlations), leading to nuisance correlations in the component existence scores. To quantify this effect, we add a component "new", and denote the argument of $\Phi(\cdot)$ in Equation (1) as A_{c} when the new component correlates with existing ones and A_{n} otherwise. Their difference can be approximated by first-order Taylor expansion:

$$A_{\text{c}} - A_{\text{n}} \approx \frac{1}{\sigma^{\text{out}}} \Big(\frac{\mu^{\text{in}} - \mu^{\text{out}}}{(\sigma^{\text{out}})^2} \sum\nolimits_j \text{Cov}^{\text{out}}_{new,j} + \frac{\Phi^{-1}(\lambda)}{\sigma^{\text{in}}} \sum\nolimits_j \text{Cov}^{\text{in}}_{new,j} - \frac{\sigma^{\text{in}} \Phi^{-1}(\lambda)}{(\sigma^{\text{out}})^2} \sum\nolimits_j \text{Cov}^{\text{out}}_{new,j} \Big), \tag{3}$$

where $\text{Cov}^{\text{in}}_{new,j}$ and $\text{Cov}^{\text{out}}_{new,j}$ denote covariance terms between the *observed existence score* of the new component and the jth existing component under ID and OOD, respectively. For *whole-image*

[3] For small $|\mathcal{P}_y|$, we provide more accurate computation in Appendix G.

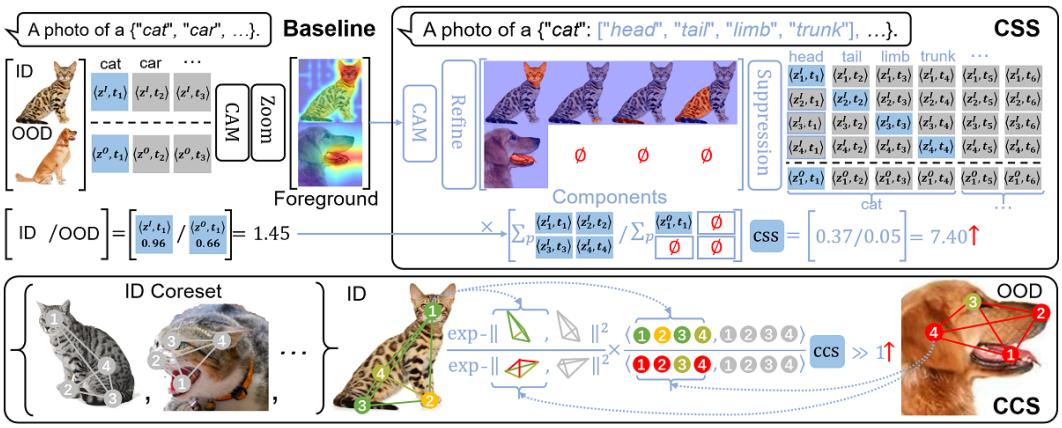

Figure 3: Illustration of CoOD framework, which firstly represents class components, and then computes complementary **CSS** and **CCS**.

inputs, the new component score may correlate non-negatively with existing ones, the *first two terms* in Equation (3) are positive. The third opposing term is typically smaller in magnitude than the second. Because ID components tend to be consistently present (low σ^{in}) and to co-occur (high $\text{Cov}^{\text{in}}_{new,j}$) in ID images. By contrast, OODs generally exhibit greater diversity (high σ^{out}) and weaker co-occurrence (low $\text{Cov}^{\text{out}}_{new,j}$). Consequently, the FPR reduction achieved by introducing additional components may be partially offset when using *full-image* input with nuisance correlations.

To address this issue, we design a text-image-feature tri-level mechanism to suppress undesired component correlations in Section 3.3. This weakens the coupling between contaminated context and the separated component existence variables, thereby reducing $\text{Cov}^{\text{in/out}}_{new,j}$ and moving the behavior of A_{c} closer to A_{n}. Empirically, this reduction in nuisance correlation mitigates cross-region interference and yields more robust component-based OOD detection.

3.3 Component identification and representation

3.3.1 Constructing a component vocabulary

A vocabulary of visually discriminative components, constructed automatically from coarse ID classes, can provide the semantic for component identification and representation. To minimize manual effort, we use LLMs to generate component texts once offline from ID class names[4].

Specifically, we prompt an LLM to decompose each class into a **"taxonomic tree"** of *visually structured component texts*, encouraging candidates to be both *class-relevant* and *semantically complete*. We then apply prompts to regularize the outputs (*e.g.* remove CLIP-insensitive modifiers such as direction or position [18], merge synonyms, and replace ambiguous or rare terms). Importantly, we explicitly allow the LLM to back off along the taxonomy tree to higher-level nodes, or adopt derivative or analogy-based decomposition when facing unreliable leaf-components. This strategy can avoid errors that arise from forcing invalid decompositions in the worst case. And the pipeline (see Appendix F for more details) aligns the extracted vocabulary with regions that a visual model is more likely to segment reliably, and thus helps ensure the discriminative component set $\mathcal{P}_y$ covers class y while reducing noise from trivial components.

3.3.2 Component localization and image-level non-component suppression

To reduce irrelevant global semantics and undesired nuisances, we suppress background and amplify component signals. Specifically, CoOD computes a foreground mask $\boldsymbol{m}_y$ via post-hoc architecture-agnostic GradCAM [39] or grounded segmentation [40], and zoom-in input to this region as Figure 3 shown. Similarly, the component masks $\boldsymbol{m}'_{yp}$ within $\boldsymbol{m}_y$ are obtained from GradCAM and compete with each other to reduce nuisance correlations, producing a refined $\boldsymbol{m}_{yp}$:

$$\boldsymbol{m}_{yp} = e^{\boldsymbol{m}'_{yp}} / (e^{1-\boldsymbol{m}_y} + \sum\nolimits_{p'} e^{\boldsymbol{m}'_{yp'}}), \tag{4}$$

[4]We also implement a purely visual, model-agnostic variant for component vocabulary construction in Section 4.3.

where $1-\boldsymbol{m}_y$ introduces a background competitor to suppress noise. Further refinement can also improve the mask quality [41]. We then obtain the component-level images by blurring $\text{Blur}(\cdot)$:

$$\boldsymbol{x}_{yp} = \boldsymbol{x} \odot \boldsymbol{m}_{yp} + \text{Blur}(\boldsymbol{x}) \odot (1 - \boldsymbol{m}_{yp}). \tag{5}$$

3.3.3 Component representation and feature-level non-component suppression

Although Equation (5) focuses on image-space components, the initial ViT class token lacks component-specific guidance and may contaminate component representations via attention. To mitigate this, we introduce a feature-level component guidance into class token and position embedding.

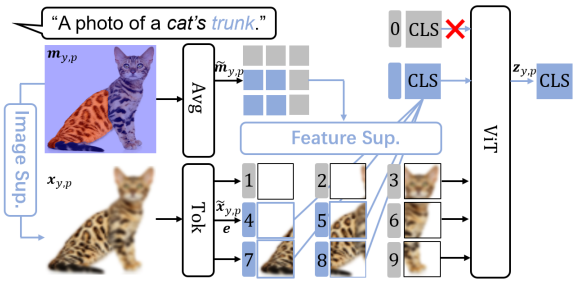

Figure 4: Illustration of image-level and feature-level non-component suppression.

As shown in Figure 4, for component $\boldsymbol{x}_{yp}$, the component mask is resized as $\boldsymbol{m}_{yp} \in [0,1]^{H\times W} \rightarrow \tilde{\boldsymbol{m}}_{yp} \in [0,1]^{N}$. Let $\tilde{\boldsymbol{x}}_{yp} = f_{\text{Tok}}(\boldsymbol{x}_{yp}) \in \mathbb{R}^{N\times D}$ be the matrix of N patch tokens, and $\boldsymbol{e}$ the position. The original patch and position tokens are concatenated to the averaged ones (*i.e.* $\frac{\tilde{\boldsymbol{m}}_{yp}^\top \tilde{\boldsymbol{x}}_{yp}}{\mathbf{1}^\top \tilde{\boldsymbol{m}}_{yp}}$ and $\frac{\tilde{\boldsymbol{m}}_{yp}^\top \boldsymbol{e}}{\mathbf{1}^\top \tilde{\boldsymbol{m}}_{yp}}$), respectively. The component feature $\boldsymbol{z}_{yp}$ is then calculated as:

$$\boldsymbol{z}_{yp} = f_{\text{ViT}}\left(\left[\tilde{\boldsymbol{m}}_{yp}^\top \tilde{\boldsymbol{x}}_{yp}/\mathbf{1}^\top \tilde{\boldsymbol{m}}_{yp}; \tilde{\boldsymbol{x}}_{yp}\right], \left[\tilde{\boldsymbol{m}}_{yp}^\top \boldsymbol{e}/\mathbf{1}^\top \tilde{\boldsymbol{m}}_{yp}; \boldsymbol{e}\right]\right), \tag{6}$$

which biases toward the target component and reduces contamination from irrelevant global semantics. These soft component guidance suppresses irrelevant information while preserving image evidence and original tokens, mitigating the impact of imperfect masks.

3.4 Component shift score

The OOD detection objective is decomposed as aggregating component-level evidences for class y:

$$P(\text{ID}\,|\,\boldsymbol{x}) = \sum\nolimits_{yp} P(\text{ID}\,|\,\boldsymbol{x}, y, p)\; P(p\,|\,\boldsymbol{x}, y)\; P(y\,|\,\boldsymbol{x}). \tag{7}$$

From [20], once the observed components $\mathcal{P}_y$ are estimated, the component-level signals explain the majority of class-relevant variation, while residual pixel-level signals provide relatively limited supplementary information. This implies a generation graph $\boldsymbol{x} \rightarrow \mathcal{P}_y \rightarrow y, \{\text{ID}, \text{OOD}\}$. Without extra prior, each component $p \in \mathcal{P}_y$ contributes equally to the ID evidence:

$$P(\text{ID}|\boldsymbol{x},y,p) \approx P(\text{ID}|y,p) = \begin{cases} 1/|\mathcal{P}_y|, & p \in \mathcal{P}_y, \\ 0, & p \notin \mathcal{P}_y. \end{cases} \tag{8}$$

Substituting Equation (8) into Equation (7) and approximating the posteriors with a neural network $P_\theta(\cdot)$, the Component Shift Score is:

$$\text{CSS}_y(\boldsymbol{x}) = P_\theta(y|\boldsymbol{x}) \frac{1}{|\mathcal{P}_y|} \sum\nolimits_{p\in\mathcal{P}_y} P_\theta(p|\boldsymbol{x}_{yp}), \tag{9}$$

where the network θ is shared for both predictions: component prediction applies suppression (y is integrated into $\boldsymbol{x}_{yp}$) as in Section 3.3, while class prediction does not. Following score-based OOD detection [1, 2], CSS admits different scoring function instantiations:

$$\text{CSS}(\boldsymbol{x}) = \max_y \frac{e^{\langle \boldsymbol{z}, \boldsymbol{t}_y \rangle}}{\sum_{y'} e^{\langle \boldsymbol{z}, \boldsymbol{t}_{y'} \rangle}} \sum_{p\in\mathcal{P}_y} \frac{e^{\langle \boldsymbol{z}_{yp}, \boldsymbol{t}_{yp} \rangle}}{|\mathcal{P}_y| \sum_{p'} e^{\langle \boldsymbol{z}_{yp}, \boldsymbol{t}_{yp'} \rangle}}. \tag{10}$$

3.5 Compositional consistency score

To detect compositional or structural OODs constructed from ID components, we design component compositional consistency score (CCS), which measures keypoint matching discrepancy to a compact

set of ID references. Given N patch tokens $\{\boldsymbol{z}_n\}_{n=1}^{N}$, the component features $\{\boldsymbol{z}_{yp}\}_{p=1}^{|\mathcal{P}_y|}$ computed in Equation (6), the corresponding masks $\{\tilde{\boldsymbol{m}}_{yp}\}_{p=1}^{|\mathcal{P}_y|}$, and let $\mathcal{I}_p(\tilde{\mathbf{m}}_{yp})$ denote the indices set of component p, we select Top$k$ indices within each component whose features are closest to the component representation $\boldsymbol{z}_{yp}$, yielding a set of keypoint indices:

$$\mathcal{I}_y(\boldsymbol{x}) = \Big\{ \underset{n\in\mathcal{I}_p(\tilde{\mathbf{m}}_{yp})}{\arg\,\text{Top}k} \; -\|\boldsymbol{z}_n - \boldsymbol{z}_{yp}\|_2, \forall p \Big\}. \tag{11}$$

To constrain the matching cost while covering ID diversity, we use a coreset $\mathcal{D}^*$ through Farthest Point Sampling [42] as ID reference. For a test $\boldsymbol{x}$ and a reference $\boldsymbol{x}^{\text{r}} \in \mathcal{D}^*$, we obtain the one-to-one keypoints matching over candidate pairs $\Pi(\mathcal{I}_y(\boldsymbol{x}), \mathcal{I}_y(\boldsymbol{x}^{\text{r}}))$:

$$\mathcal{K}_y^*(\boldsymbol{x}, \boldsymbol{x}^{\text{r}}) = \underset{\mathcal{K}\in\Pi(\mathcal{I}_y(\boldsymbol{x}),\mathcal{I}_y(\boldsymbol{x}^{\text{r}}))}{\arg\max} \sum_{(i,j)\in\mathcal{K}} \langle \boldsymbol{z}_i, \boldsymbol{z}_j^{\text{r}} \rangle. \tag{12}$$

We then define CCS as the distance-weighted feature similarity over the keypoints matching:

$$\text{CCS}(\boldsymbol{x}) = \max_{\boldsymbol{x}^{\text{r}}} \sum_{(i,j)\in\mathcal{K}_y^*(\boldsymbol{x},\boldsymbol{x}^{\text{r}})} e^{-\|\mathcal{M}(\boldsymbol{e}_i)-\boldsymbol{e}_j\|^2} \langle \boldsymbol{z}_i, \boldsymbol{z}_j^{\text{r}} \rangle / |\mathcal{K}_y^*(\boldsymbol{x}, \boldsymbol{x}^{\text{r}})|, \tag{13}$$

where $\mathcal{M}$ is a geometric registration based on the matched $\mathcal{K}_y^*(\boldsymbol{x}, \boldsymbol{x}^{\text{r}})$ that provides approximate spatial correspondence. In the CoOD implementation, we use a classical homography-based alignment as a practical choice. While simple, such approximate alignment is often sufficient to detect substantial compositional inconsistency: the diverse coreset reference ensures coverage of ID variations beyond homography. This observation is consistent with previous work [19, 43], which suggests that IDs tend to retain more coherent correspondences than OODs after approximate alignment. The distance decay term thus amplifies sensitivity to geometric mismatch, while the feature similarity captures semantic agreement. Together, CCS detects compositional deviations that are invisible to CSS alone.

4 Experiments

4.1 Experiment setup

Datasets. We primarily evaluate CoOD under 3 settings: **(i) ImageNet1K** [44] in CLIP protocol, where SUN [45], Places [46], iNaturalist [47], and DTD [48] are OOD datasets; **(ii) Fine-grained CUB** [49], where half of the subclasses within each superclass are used as ID and the remaining subclasses, together with other coarse classes, are treated as OOD; **(iii) Compositional OOD on counterfactual data and ImageNet splits**, which isolate compositional changes via two constructions: **(iii)** generating counterfactual samples using a diffusion model (FLUX.2[klein]9B [50]) with prompt *"A surreal industrial design of a* $\langle class \rangle$*, where* $\langle component \rangle$ *mounted on* ***impossible*** *position."*, where paired IDs use the *normal* position and same seed, preserving comparable non-compositional distributions; and **(iii')** manually regrouping ImageNet classes that share similar components but exhibit compositional inconsistencies. Note that we use the same preprocessing and balance sampling for a fair comparison. More details and benchmarks are provided in the Appendix.

Implementation Details. We generate the components text by ChatGPT-4o (see Appendix F), encode features by CLIP ViT-B/16, and obtain visual components by GradCAM [39] with mask refinement [51, 52], without training CLIP-ES [39]. CoOD only requires localizable component features and their representations, rather than a specific encoder. Algorithm 1 provides our pseudo code for CoOD. Moreover, we design a variant CoOD* that uses the OOD labels as in EOE [26] to refine CSS probability, and a faster version of CoOD-F that replaces component features z_{yp} with the global z (see Appendix D). CoOD uses coreset with 20 reference ($\sim 1\%|\mathcal{D}|$), temperature τ=1, keypoints k=4 to satisfy the freedom degrees for spatial transformation, and selects the weight α on a validation set (pseudo OODs could be generated by rotation-90 and gaussian blurring, the empirical values are discussed in Section 4.3).

All compared methods are listed in Appendix B, where most tuning-based methods follow the original papers of [9] without additional regularization (50 epoch, 0.002 learning rate, 32 batch size, 1 warm-up epoch, 20-shot, 16 token lengths), while IDlike and LoPro follow their best setting in [6] and [14]. Their temperature τ is selected from the best of $\{0.01, 1\}$. We report macro-average area under the receiver operating characteristic curve (AUC) and false positive rate at 95% recall (FPR).

Table 1: OOD detection on ImageNet-1K and CUB with various OOD detectors and models. **Boldface** indicates the best performance. * indicates methods that require OOD labels (we use the original ID-only version of Dual and ΔE for a fair comparison with CoOD).

$\mathcal{D}$	Metr.	Tr.	MLS	KL	EBO	ViM	ODIN	MCM	Gap	GL/R	EOE*	NEG*	Dual	CSP*	COOK	ΔE	CoOD-F	CoOD	CoOD*
ImageNet-1k	AUC	CLIP	84.6	74.9	81.5	61.9	90.4	90.2	89.3	91.0	93.3	94.2	90.9	95.7	90.8	90.3	95.4	96.2	**97.4**
		CoOp	84.8	72.9	81.6	63.2	90.5	91.9	88.1	92.6	92.3	91.0	85.4	92.6	89.8	86.3	96.2	96.7	**97.6**
		LoCoOp	79.8	75.6	73.2	60.4	90.5	92.7	89.0	93.6	93.2	86.4	86.5	87.9	90.0	88.1	96.3	96.9	**97.8**
		IDLike	83.0	78.6	81.1	70.3	90.3	91.2	90.5	92.7	89.6	86.2	89.5	88.8	91.7	90.4	96.3	96.8	**96.9**
		LoPro	84.5	74.6	81.7	64.5	90.4	92.6	89.4	93.6	93.4	89.8	90.0	93.2	90.9	90.4	95.4	96.2	**97.4**
	FPR_{95}	CLIP	46.7	66.2	49.2	69.4	41.4	37.6	40.1	35.7	26.2	25.4	39.5	22.3	35.9	35.6	17.9	15.6	**11.8**
		CoOp	53.8	67.8	57.1	70.5	41.4	36.1	46.0	30.0	34.6	40.2	53.4	34.3	41.0	50.3	14.2	12.4	**10.7**
		LoCoOp	63.6	64.5	69.8	77.9	41.4	34.0	44.5	28.7	31.0	50.9	52.0	47.5	40.5	47.7	14.0	11.8	**10.1**
		IDLike	52.6	60.8	54.3	64.0	41.2	37.7	40.9	38.0	38.2	42.4	43.5	36.6	35.9	38.9	14.0	13.0	**12.1**
		LoPro	47.0	70.7	49.2	67.1	41.4	37.3	39.7	28.3	25.7	32.2	38.6	26.7	35.6	35.6	17.7	15.7	**11.8**
CUB	AUC	CLIP	77.5	65.2	76.9	63.4	74.2	75.4	67.5	73.5	77.8	76.3	76.4	76.5	75.6	76.8	77.4	85.2	**86.7**
		CoOp	81.9	63.1	81.5	65.8	78.1	80.4	78.8	78.2	82.1	78.1	81.3	78.3	79.4	76.8	82.5	87.7	**88.3**
		LoCoOp	78.3	60.0	76.1	64.1	79.2	80.7	80.5	78.3	82.5	74.2	82.0	79.3	79.3	76.8	83.2	88.5	**89.6**
		IDLike	80.9	65.3	80.5	65.7	75.2	76.2	70.2	75.5	73.9	79.0	79.2	80.8	75.3	76.6	82.7	87.9	**89.0**
		LoPro	77.0	65.3	76.3	63.4	74.5	75.2	71.9	75.9	77.5	75.5	76.1	76.1	75.2	74.3	77.7	84.5	**85.0**
	FPR_{95}	CLIP	71.2	81.6	71.3	83.8	76.4	73.0	74.0	76.3	73.0	74.8	70.5	83.5	72.7	76.2	69.7	56.5	**54.4**
		CoOp	65.8	86.7	66.7	78.5	73.0	67.8	66.6	73.4	67.3	72.2	66.7	68.4	69.1	74.0	62.2	**49.3**	51.0
		LoCoOp	68.0	88.1	73.1	82.1	67.3	64.6	63.1	67.2	62.1	76.4	62.1	67.6	66.3	74.0	54.7	45.9	**40.3**
		IDLike	66.5	81.4	68.6	80.6	76.6	73.9	70.5	74.9	77.2	76.4	69.4	67.8	74.2	73.7	59.6	**45.5**	46.0
		LoPro	71.3	82.0	73.0	82.7	76.2	73.9	76.7	72.0	70.5	75.7	72.3	85.0	74.2	77.1	69.4	59.6	**55.4**

Table 2: Compositional OOD detection on ImageNet-splits and generated counterfactual dataset.

Metr.	$\mathcal{D}$	MLS	KL	EBO	ViM	ODIN	MCM	Gap	GL	EOE*	NEG*	Dual	CSP*	COOK	ΔE	CoOD-F	CoOD	CoOD*
AUC	ImageNet-Split	68.3	51.4	65.5	55.6	68.0	68.8	69.9	69.7	68.6	68.4	70.0	68.4	56.3	68.4	64.6	75.4	**76.0**
	Counterfact	59.6	39.7	59.1	76.3	52.5	54.5	57.9	54.6	51.3	55.6	57.5	55.7	58.3	50.3	81.1	**86.3**	85.1
FPR	ImageNet-Split	78.7	94.0	82.1	89.3	86.2	78.0	76.4	77.5	74.9	75.5	78.7	76.3	88.1	75.8	83.4	**66.7**	69.1
	Counterfact	92.0	90.2	93.8	**55.4**	94.6	95.5	92.9	94.5	99.1	92.9	93.5	92.0	81.3	83.9	58.0	62.5	62.5

4.2 Main results

Component-level evidence provides consistent improvements. Tables 1 and 7 to 10 report the results of CoOD and its variants on coarse-grained ImageNet. Across a wide range of detectors, models, and evaluation settings, including challenging near-OOD in Table 9 and complex scenarios in Table 10, CoOD consistently outperforms both traditional OOD detectors and recent CLIP-based methods, suggesting that component-level evidence improves detection reliability. The faster variant without explicit visual component extraction (CoOD-F) also achieves competitive performance and surpasses methods that do not use OOD labels, though below the full version of CoOD. Moreover, when combined with auxiliary OOD labels, CoOD* is further improved, indicating compatibility with existing OOD-exposure strategies. The statistical significance is reported in Appendix E.5.

CoOD is particularly effective for fine-grained and compositional OOD detection. As shown in Tables 1 and 2, CoOD yields a substantial performance gain on fine-grained and compositional OOD benchmarks relative to strong baselines. In particular, fine-grained appearance variations (*e.g.*, bird plumage or leg in CUB) are captured by CSS, whereas compositional changes (*e.g.*, placing a handle inside a cup in Figure 6) are captured by CCS. Taken together, these results indicate that CoOD can address distribution shifts across different granularities and compositional structures.

4.3 Analysis

Non-textual CoOD. When LLMs are unavailable, we implement CoOD with a **visual-vocabulary variant** by super-pixel clustering [53] and feed them into the same pipeline, reaching only slightly worse results (-0.5/-0.1 AUC +1.8/+0.9 FPR on ImageNet and -2.4/-3.0 AUC +9.5/+15.1 FPR on CUB). Although language can aid in semantic alignment, the principal gains of CoOD stem from its component-centric design rather than reliance on LLMs.

Effectiveness of Proposed Modules. Table 3 summarizes ablations for our key modules on ImageNet-54. **CCS** supplies compositional cues, but without **Zoom** processing, the keypoint sampling and matching is noisy due to low foreground resolution and background interference. For **CSS**, extracting components without **Sup**pression degrades performance, consistent with the increased nuisance

Table 3: Ablation study of proposed modules (AUC/FPR).

CSS	Baseline		✓	✓	✓	✓	✓	✓
CCS		✓						✓
Sup		-	-		Post-Agg	✓	✓	✓
Zoom			✓	-			✓	✓
AUC/FPR	93.5/24.0	95.2/16.2	96.7/14.2	94.8/19.8	94.9/19.0	95.9/16.1	96.6/14.3	**98.4/9.7**

Table 4: Ablation study of component number $|\mathcal{P}_y|$.

$\lvert\mathcal{P}_y\rvert$	0	3	4*	5	12
AUC	75.4	86.0	**86.7**	86.1	85.7
FPR	73.0	**49.6**	54.4	59.6	66.6

Table 5: Ablation study of coreset size $|\mathcal{D}|$ in CCS.

$\lvert\mathcal{D}\rvert$	1	0.1%	0.2%	1%
AUC	97.5	98.0	98.4	**98.8**
FPR	10.9	8.8	7.1	**5.9**

Table 6: Ablation study of the weight factor α. We report results of CoOD / CoOD* for each setting.

α	Coarse-Grained OOD					Fine-Grained OOD				
	0.1	0.25	0.5*	0.75	1	0.1	0.2*	0.3	0.4	0.5
AUC	97.1 / 98.4	97.7 / 98.8	**98.0 / 98.9**	**98.0** / 98.7	97.8 / 98.5	84.9 / 86.6	**85.2 / 86.7**	82.7 / 84.9	81.1 / 83.5	79.2 / 81.8
FPR	12.9 / 7.3	10.0 / **5.8**	**9.9 / 5.8**	**9.9** / 6.5	10.7 / 7.5	**55.6 / 53.7**	56.5 / 54.4	60.0 / 55.3	63.5 / 58.8	66.9 / 61.7

correlation in Equation (3), context leakage, and background noise. A alternative that averages the final component tokens (Post-Agg) is also ineffective, suggesting that early-stage contamination cannot be suppressed at the final layer. After suppression, **Zoom** further improves the component quality. Combining **CSS** with **CCS** yields the best results, confirming their complementary benefits.

Component vocabulary robustness. Appendix F shows that the generated vocabulary has low empirical error rates under our prompt. Even under an adversarial prompt *"each class must include at least 1* ***rare or spurious component****"* that injects problematic components, CoOD maintains nearly identical performance (ImageNet, -0.1 AUC / $+0.6$ FPR). Moreover, we repeat the LLM generation process for 3 times to assess the standard deviation (0.06 AUC / 0.36 FPR). These results demonstrate the robustness of CoOD to spurious components and generation stochasticity.

We further study component number via standard CUB annotation in Table 4. With few components, LLM captures salient parts (*e.g.*, bird head) and yields strong gains, more efficient than using thousands of OOD labels. As component number grows and covers the visual foreground, performance saturates due to the limited encoder resolution that components at this granularity cannot be reliably encoded. To avoid introducing potentially unreliable vocabulary, our prompts allow fallback to higher-level concepts or analogy-based decomposition as in Appendix F, improving the robustness.

Coreset size. We study the sensitivity of the coreset strategy by varying the size $|\mathcal{D}^*| \in \{1,2(0.1\times),4(0.2\times),200(10\times)\}$ on ImageNet-54. A larger coreset improves the coverage of rare ID samples and bolstering detection performance, as evidenced in Table 5. On the other hand, the marginal differences verify the effectiveness and robustness of our coreset strategy.

Weight α. Table 6 indicates that CoOD is relatively stable to weight α. Take a closer look, a relatively small $\alpha = 0.2$ is slightly better for fine-grained CUB, as these tasks typically involve minimal geometric changes within the same superclass. In contrast, a larger $\alpha = 0.5$ benefits coarse-grained detection in ImageNet-54, since it identifies global OODs through compositional inconsistencies.



Figure 5: Illustration of challenging OODs. The percentages represent ranking percentiles, with lower values indicating more probable OODs. The local responses [14] are highlighted.

Visualization. Figure 5 shows two challenging OOD samples that traditional methods cannot detect. Global detector mis-classifies the two samples (whose ground truth are "crane barge" and "blue grosbeak") to ID classes due to high visual similarity. Local method cannot output correct results either, as it produces noisy responses and misses long-range relations. For

CoOD, the first sample lacks the components of “drilling-platform” thus receives a high component shift, and the spatial misalignments with ID compositions are also flagged by CCS. In the second case, CoOD localizes the components corresponding to “indigo bunting” (visually similar to “blue grosbeak”) and suppresses the orange-feather signal on the “wing” which is absent from the ID component, thereby correctly detects this fine-grained OOD.

Notably, the "support" and "deck" components are imperfectly localized in the crane barge example. However, this may arise from the absence of corresponding ID components rather than localization errors alone, and thus has limited impact on the final OOD decision in this case.

Efficiency. We run experiments on an EPYC 9654-RTX 4090 server and report per-image inference time: MCM, CoOD-F, and CoOD(*) achieve 89, 85, and 1.5 FPS. Component representation, CSS, and CCS contribute 72%, 2%, and 26% of the total cost. A fixed-size ID coreset limits the reference-matching overhead. The lower throughput of full CoOD results from its training-free, model-agnostic design, which avoids specialized segmentation or registration modules. This introduces an accuracy-efficiency trade-off, with substantial gains on fine-grained and compositional OOD detection.

5 Conclusion

We present Component-based OOD Detection (CoOD), a post-hoc framework that (1) identifies and aggregates patches into components to suppress noise and expose fine-grained OODs (Component Shift Score, CSS), and (2) evaluates compositional OOD among components to reveal invalid compositions (Compositional Consistency Score, CCS). CoOD provides inspectable component-level evidence and consistently improves both coarse and fine-grained OOD detection. Future work will extend spatial components to more general feature-level components, such as color and viewpoint, providing a realistic pathway toward more interpretable OOD detection in real-world settings.

6 Broader Impacts

Fine-grained distributional shifts are common in real-world deployments, and practitioners frequently require interpretable, component-level evidence to better understand or validate OOD decisions. This paper introduces a component-centric, post-hoc paradigm for OOD detection that reconciles the trade-off between sensitivity to fine-grained OODs and robustness to natural ID variability. Our method detects subtle appearance and compositional OODs without retraining, is broadly applicable to practical vision systems, and improves the safety and reliability of downstream decision-making. More broadly, this work highlights the potential of representation learning grounded solely in training distributions and strategies to model existing data more faithfully, fostering progress in building trustworthy AI. However, in safety-critical applications, missed OODs or false alarms may lead to unreliable decisions or unnecessary abstentions, requiring appropriate validation before deployment.

7 Limitations

A practical limitation is that the final number of components may be lower than the initially intended granularity. Small or ambiguous regions may require coarser decomposition, sacrificing detail for reliability. CoOD primarily targets object-centric semantic OOD with identifiable components, while CCS additionally requires meaningful spatial relations. Its applicability to texture-dominated or other non-object-centric domains remains unverified. Incorrect vocabularies, poor masks, occlusion, unusual viewpoints, and multiple salient objects may degrade performance, which soft suppression cannot fully prevent.

The standard CoOD exhibits a relatively low throughput due to its post-hoc and model-agnostic design, relying on classical matching and localization steps. For efficiency-sensitive scenarios, our faster variant CoOD-F achieves throughput comparable to existing baselines while retaining strong performance. Nevertheless, full CoOD remains unsuitable for latency-sensitive applications without further acceleration through efficient auxiliary modules. Finally, LLM-based vocabulary generation introduces concerns regarding reliance on proprietary APIs, although such practices are increasingly common in the field and visual or lexical alternatives are available.

Acknowledgments and Disclosure of Funding

This work is supported by National Natural Science Foundation of China (No. 62495082, 62376259).

References

[1] Weitang Liu, Xiaoyun Wang, John D. Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems*, 2020.

[2] Yifei Ming, Ziyang Cai, Jiuxiang Gu, Yiyou Sun, Wei Li, and Yixuan Li. Delving into out-of-distribution detection with vision-language representations. In *Advances in Neural Information Processing Systems*, 2022.

[3] Jiachen Liang, Ruibing Hou, Minyang Hu, Hong Chang, Shiguang Shan, and Xilin Chen. Revisiting logit distributions for reliable out-of-distribution detection. In *Advances in Neural Information Processing Systems*, 2025.

[4] Kihyuk Sohn, Chun-Liang Li, Jinsung Yoon, Minho Jin, and Tomas Pfister. Learning and evaluating representations for deep one-class classification. In *International Conference on Learning Representations*, 2021.

[5] Jinjing Hu, Wenrui Liu, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. An information theoretical view for out-of-distribution detection. In *European Conference on Computer Vision*, volume 15113, pages 418–435, 2024.

[6] Yichen Bai, Zongbo Han, Bing Cao, Xiaoheng Jiang, Qinghua Hu, and Changqing Zhang. Id-like prompt learning for few-shot out-of-distribution detection. In *Computer Vision and Pattern Recognition*, pages 17480–17489, 2024.

[7] Tal Reiss, Niv Cohen, and Yedid Hoshen. Is scale all you need for anomaly detection?, 2024.

[8] Atsuyuki Miyai, Qing Yu, Go Irie, and Kiyoharu Aizawa. Zero-shot in-distribution detection in multi-object settings using vision-language foundation models. *arXiv:2304.04521*, 2023.

[9] Atsuyuki Miyai, Qing Yu, Go Irie, and Kiyoharu Aizawa. Locoop: Few-shot out-of-distribution detection via prompt learning. In *Advances in Neural Information Processing Systems*, 2023.

[10] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: A good closed-set classifier is all you need. In *International Conference on Learning Representations*, 2022.

[11] Yifei Ming, Hang Yin, and Yixuan Li. On the impact of spurious correlation for out-of-distribution detection. In *AAAI Conference on Artificial Intelligence*, 2022.

[12] Katie Kang, Amrith Setlur, Claire J. Tomlin, and Sergey Levine. Deep neural networks tend to extrapolate predictably. In *International Conference on Learning Representations*, 2024.

[13] Ziqian Lin, Sreya Dutta Roy, and Yixuan Li. MOOD: multi-level out-of-distribution detection. In *Computer Vision and Pattern Recognition*, pages 15313–15323, 2021.

[14] Fanhu Zeng, Zhen Cheng, Fei Zhu, Hongxin Wei, and Xu-Yao Zhang. Local-prompt: Extensible local prompts for few-shot out-of-distribution detection. In *International Conference on Learning Representations*, 2025.

[15] Mohammad Pezeshki, Amartya Mitra, Yoshua Bengio, and Guillaume Lajoie. Multi-scale feature learning dynamics: Insights for double descent. In *International Conference on Machine Learning*, volume 162, pages 17669–17690, 2022.

[16] Bum Jun Kim, Hyeyeon Choi, Hyeonah Jang, and Sang Woo Kim. Understanding gaussian attention bias of vision transformers using effective receptive fields. In *British Machine Vision Conference*, page 214, 2023.

[17] Sara Sabour, Nicholas Frosst, and Geoffrey E. Hinton. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems*, pages 3856–3866, 2017.

[18] Wenliang Dai, Zihan Liu, Ziwei Ji, Dan Su, and Pascale Fung. Plausible may not be faithful: Probing object hallucination in vision-language pre-training. In *European Chapter of the Association for Computational Linguistics*, pages 2128–2140, 2023.

[19] Wenrui Liu, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Diversity-measurable anomaly detection. In *Computer Vision and Pattern Recognition*, pages 12147–12156, June 2023.

[20] Irving Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94 2:115–147, 1987.

[21] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017.

[22] Lin Zhu, Yifeng Yang, Xinbing Wang, Qinying Gu, and Nanyang Ye. Delta energy: Optimizing energy change during vision-language alignment improves both ood detection and ood generalization. In *Advances in Neural Information Processing Systems*, 2025.

[23] Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In *International Conference on Machine Learning*, volume 162, pages 20827–20840, 2022.

[24] Galadrielle Humblot-Renaux, Gianni Franchi, Sergio Escalera, and Thomas B. Moeslund. Cooked: Ensemble-based OOD detection in the era of zero-shot CLIP. In *International Conference on Computer Vision Workshop*, pages 261–271. IEEE, 2025.

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763, 2021.

[26] Chentao Cao, Zhun Zhong, Zhanke Zhou, Yang Liu, Tongliang Liu, and Bo Han. Envisioning outlier exposure by large language models for out-of-distribution detection. In *International Conference on Machine Learning*, 2024.

[27] Xue Jiang, Feng Liu, Zhen Fang, Hong Chen, Tongliang Liu, Feng Zheng, and Bo Han. Negative label guided OOD detection with pretrained vision-language models. In *International Conference on Learning Representations*, 2024.

[28] Mengyuan Chen, Junyu Gao, and Changsheng Xu. Conjugated semantic pool improves OOD detection with pre-trained vision-language models. In *Advances in Neural Information Processing Systems*, 2024.

[29] Ji Zhang, Lianli Gao, Bingguang Hao, Hao Huang, Jingkuan Song, and Hengtao Shen. From global to local: Multi-scale out-of-distribution detection. *IEEE Transactions on Image Processing*, 32:6115–6128, 2023.

[30] Fayi Le, Wenwu He, Chentao Cao, Dong Liang, and Zhuo-Xu Cui. Dualcnst: Enhancing zero-shot out-of-distribution detection via text-image consistency in vision-language models. In *Advances in Neural Information Processing Systems*, 2025.

[31] Sen Pei. Image background serves as good proxy for out-of-distribution data. In *International Conference on Learning Representations*, 2024.

[32] Dan Hendrycks, Mantas Mazeika, and Thomas G. Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2019.

[33] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan Su. This looks like that: Deep learning for interpretable image recognition. In *Advances in Neural Information Processing Systems*, pages 8928–8939, 2019.

[34] Ronan Sicre, Hanwei Zhang, Julien Dejasmin, Chiheb Daaloul, Stéphane Ayache, and Thierry Artières. Dp-net: Learning discriminative parts for image recognition. In *International Conference on Image Processing*, pages 1230–1234, 2023.

[35] Zhijie Zhu, Lei Fan, Maurice Pagnucco, and Yang Song. Interpretable image classification via non-parametric part prototype learning. In *Computer Vision and Pattern Recognition*, pages 9762–9771, 2025.

[36] Adam Kortylewski, Qing Liu, Angtian Wang, Yihong Sun, and Alan L. Yuille. Compositional convolutional neural networks: A robust and interpretable model for object recognition under occlusion. *International Journal of Computer Vision*, 129(3):736–760, 2021.

[37] Xiao Li, Ziqi Wang, Bo Zhang, Fuchun Sun, and Xiaolin Hu. Recognizing object by components with human prior knowledge enhances adversarial robustness of deep neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7):8861–8873, 2023.

[38] Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. The pitfalls of simplicity bias in neural networks. In *Advances in Neural Information Processing Systems*, 2020.

[39] Yuqi Lin, Minghao Chen, Wenxiao Wang, Boxi Wu, Ke Li, Binbin Lin, Haifeng Liu, and Xiaofei He. CLIP is also an efficient segmenter: A text-driven approach for weakly supervised semantic segmentation. In *Computer Vision and Pattern Recognition*, pages 15305–15314, 2023.

[40] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, RISHI HAZRA, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollar, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. Sam 3: Segment anything with concepts. In *International Conference on Learning Representations*, volume 2026, pages 138846–138923, 2026.

[41] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in Neural Information Processing Systems*, pages 109–117, 2011.

[42] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, pages 5099–5108, 2017.

[43] Jie Ren, Peter J. Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark A. DePristo, Joshua V. Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. In *Advances in Neural Information Processing Systems*, pages 14680–14691, 2019.

[44] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[45] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *Computer Vision and Pattern Recognition*, pages 3485–3492, 2010.

[46] Bolei Zhou, Àgata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464, 2018.

[47] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alexander Shepard, Hartwig Adam, Pietro Perona, and Serge J. Belongie. The inaturalist species classification and detection dataset. In *Computer Vision and Pattern Recognition*, pages 8769–8778, 2018.

[48] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Computer Vision and Pattern Recognition*, pages 3606–3613, 2014.

[49] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, Jul 2011.

[50] Black Forest Labs. FLUX.2: Frontier Visual Intelligence. https://bfl.ai/blog/flux-2, 2025.

[51] Sun Shuyang, Li Runjia, Torr Philip, Gu Xiuye, and Li Siyang. Clip as rnn: Segment countless visual concepts without training endeavor. In *Computer Vision and Pattern Recognition*, pages 13171–13182, 2024.

[52] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *International Conference on Computer Vision*, pages 4015–4026, 2023.

[53] Amirata Ghorbani, James Wexler, James Zou, and Been Kim. Towards automatic concept-based explanations. In *Advances in Neural Information Processing Systems*, 2019.

[54] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.

[55] Dan Hendrycks, Steven Basart, Mantas Mazeika, Andy Zou, Joseph Kwon, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. In *International Conference on Machine Learning*, volume 162, pages 8759–8773, 2022.

[56] Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. Vim: Out-of-distribution with virtual-logit matching. In *Computer Vision and Pattern Recognition*, pages 4911–4920, 2022.

[57] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018.

[58] Xixi Liu and Zach Christopher. Tag: Text prompt augmentation for zero-shot out-of-distribution detection. In *European Conference on Computer Vision*, 2024.

[59] Geng Yu, Jianing Zhu, Jiangchao Yao, and Bo Han. Self-calibrated tuning of vision-language models for out-of-distribution detection. In *Advances in Neural Information Processing Systems*, 2024.

[60] Wenjun Miao, Guansong Pang, Zihan Wang, Jin Zheng, and Xiao Bai. Auxiliary prompt tuning of vision-language models for few-shot out-of-distribution detection. In *International Conference on Computer Vision*, pages 1–10. IEEE, 2025.

[61] Zhuo Xu, Xiang Xiang, and Yifan Liang. Overcoming shortcut problem in VLM for robust out-of-distribution detection. In *Computer Vision and Pattern Recognition*, pages 15402–15412, 2025.

[62] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. In *Advances in Neural Information Processing Systems*, pages 9448–9458, 2019.

A Compositional OOD setup

A.1 The ImageNet compositional OOD splits

The compositional OOD splits of ImageNet are manually constructed by grouping classes that share similar visual components and differ mainly in composition. Specifically, we identify the following subgroups as more likely to exhibit shared components and different compositional variations: `['barber chair', 'folding chair', 'rocking chair']`, `['dining table', 'desk']`, `['bicycle-built-for-two', 'tricycle', 'unicycle', 'mountain bike']`, `['warplane', 'airliner']`, `['rifle', 'assault rifle']`, `['ladle', 'wooden spoon']`, `['mixing bowl', 'soup bowl']`, `['teapot', 'pitcher']`, `['water bottle', 'beer glass', 'measuring cup', 'red wine']`, `['cup', 'coffee mug']`. Within each subgroup, we systematically rotate the ID assignment such that each class is designated as ID in turn, with all other subclasses regarded as OOD during that iteration.

A.2 The counterfactual data

While this ImageNet split provides a benchmark for compositional OOD evaluation, it does not fully control for component consistency and often entangles compositional variation with other nuisance factors. To isolate compositional changes, we further generate one-component-changed counterfactual samples using a diffusion model (FLUX.2-9B [50], $guidance = 5$, $size = 512 \times 512$) conditioned on prompts of the form: *"A surreal industrial design prototype of a* $\langle class \rangle$*, where* $\langle component \rangle$ *mounted on* ***impossible*** *position."*. To ensure the validity of these prompts, we use classes from artifacts in ImageNet-1K that possess at least four distinct semantic components. We then use the corresponding prompt *"A surreal industrial design prototype of a* $\langle class \rangle$*, where* $\langle component \rangle$ *mounted on* ***normal*** *position."* to generate ID data. By using matched prompts and identical random seeds, we aim to isolate compositional changes while minimizing unrelated visual variations. Figure 6 provides representative examples of the generated counterfactual dataset.



Figure 6: Illustration of generated counterfactual dataset for testing compositional OODs.

B Compared methods

We focus primarily on CLIP-based methods, where the models (training-based methods) include: zero-shot inference (CLIP) [2], prompt learning (CoOp) [54], local regularization (LoCoOp) [9], ID-like prompt (IDLike) [6], and local prompt (LoPro) [14].

The detectors are detached from training tricks, including traditional output space methods (the first group in our main results): MaxLogit (MLS) [55], energy (EBO) [1], Kullback–Leibler divergence (KL) [55], dominant-component score (ViM) [56] and input perturbation (ODIN) [57]; and CLIP-based methods (the second group in our main results): maximum concept matching (MCM) [2], local enhancement (GL) [8] / (R) [14], energy change (ΔE) [22] ensemble learning (COOK) [24], visual similarity (Dual) [30], OOD label grouping/exposure/semantic-pool (NEG/EOE/CSP) [26–28], text prompt augmentation (TAG) [58] and logit aggregation (Gap) [3]. Note that all method kept consistent with their original parameters unless explicitly stated otherwise, and methods require OOD labels are highlighted by *.

Related local-global methods include SCT [59], APT [60], and OSPCoOp [61]. SCT calibrates the influence of background-derived OOD regularization during prompt tuning, APT exploits foreground/background OOD features from auxiliary outliers, and OSPCoOp mitigates background shortcuts through mask-guided regularization. Unlike these training-based methods, CoOD explicitly models within-object components and their geometric compositions in a post-hoc manner.

C Pseudo code of CoOD

We provide the pseudo code of CoOD as follows:

Algorithm 1 Component-based OOD Detection

Input: testing image $\boldsymbol{x}$, class labels $\mathcal{T}$, class components $\mathcal{P}=\text{LLM}(\mathcal{T})$, coreset data $\mathcal{D}^*=\text{FPS}(\mathcal{D})$

Output: CoOD scores $CSS(\boldsymbol{x})$ and $CCS(\boldsymbol{x})$ for given testing image $\boldsymbol{x}$

Component shift score $CSS(\boldsymbol{x})$
Compute textual class features $\quad \boldsymbol{t}_y \;= g(\mathcal{T})$
Compute textual component features $\quad \boldsymbol{t}_{yp} \;= g(\mathcal{P})$
Compute visual class features $\quad \boldsymbol{z} \;= f(\boldsymbol{x})$

Compute class mask $\quad \boldsymbol{m}_y \;= \text{GradCAM}(\boldsymbol{z}; \boldsymbol{t}_y)$
Zoom $\boldsymbol{x}$ to foreground $\quad \boldsymbol{x} \;= \text{Zoom}(\boldsymbol{x}; \boldsymbol{m}_y)$
Compute component masks $\quad \boldsymbol{m}_{yp} = \text{GradCAM}(x; \boldsymbol{t}_{yp})$
Refine component masks $\boldsymbol{m}_{yp}$ by Equation (4) using $\boldsymbol{m}_y$ and $\boldsymbol{m}_{yp}$

Compute visual component features $\quad \boldsymbol{z}_{yp} \;= f(\text{Suppress}(\boldsymbol{x}; \boldsymbol{m}_{yp}))$
Compute $CSS(\boldsymbol{x})$ by Equation (10)

Compositional consistency score $CCS(\boldsymbol{x})$
Sample references $\boldsymbol{x}^{\text{r}}$ from $\mathcal{D}^*$
Compute visual reference features $\quad \boldsymbol{z}^{\text{r}} \;= f(\boldsymbol{x}^{\text{r}})$
Sample keypoints $\mathcal{I}_y(\boldsymbol{x})$, $\mathcal{I}_y(\boldsymbol{x}^{\text{r}})$ by Equation (11) using $\boldsymbol{z}$, $\boldsymbol{z}^{\text{r}}$, and $\boldsymbol{z}_{yp}$
Match keypoints $\mathcal{K}^*_y(\boldsymbol{x}, \boldsymbol{x}^{\text{r}})$ by Equation (12) using $\boldsymbol{z}$, $\boldsymbol{z}^{\text{r}}$, $\mathcal{I}_y(\boldsymbol{x})$, and $\mathcal{I}_y(\boldsymbol{x}^{\text{r}})$
Compute $CCS(\boldsymbol{x})$ by Equation (13)

D The variants of CoOD

We introduce a computationally efficient variant, CoOD-F, by replacing the component-level features $\boldsymbol{z}_{yp}$ with the global feature $\boldsymbol{z}$:

$$\text{CSS-F}(\boldsymbol{x}) = \max_y \frac{e^{\langle \boldsymbol{z}, \boldsymbol{t}_y \rangle}}{\sum_{y'} e^{\langle \boldsymbol{z}, \boldsymbol{t}_{y'} \rangle}} \sum_{p \in \mathcal{P}_y} \frac{e^{\langle \boldsymbol{z}, \boldsymbol{t}_{yp} \rangle}}{|\mathcal{P}_y| \sum_{p'} e^{\langle \boldsymbol{z}, \boldsymbol{t}_{yp'} \rangle}}. \tag{14}$$

Furthermore, our framework can be naturally extended to incorporate negative-label-based methods, denoted as CoOD*. For instance, following EOE [26], we can utilize the same negative labels $\boldsymbol{t}_{y^-} \in \{g(\text{“text}_{y^-}\text{”})\}_{y^-=1}^{C^-}$ to refine the CSS probability with a default weight $\beta = 0.5$:

$$\text{CSS}^*(\boldsymbol{x}) = \max_y \frac{e^{\langle \boldsymbol{z}, \boldsymbol{t}_y \rangle} - \beta \max_{y^-} e^{\langle \boldsymbol{z}, \boldsymbol{t}_{y^-} \rangle}}{\sum_{y'} e^{\langle \boldsymbol{z}, \boldsymbol{t}_{y'} \rangle} + \sum_{y'^-} e^{\langle \boldsymbol{z}, \boldsymbol{t}_{y'^-} \rangle}} \sum_{p \in \mathcal{P}_y} \frac{e^{\langle \boldsymbol{z}_{yp}, \boldsymbol{t}_{yp} \rangle} - \beta \max_{y^-} e^{\langle \boldsymbol{z}, \boldsymbol{t}_{y^-} \rangle}}{|\mathcal{P}_y| \left(\sum_{p'} e^{\langle \boldsymbol{z}_{yp}, \boldsymbol{t}_{yp'} \rangle} + \sum_{y'^-} e^{\langle \boldsymbol{z}, \boldsymbol{t}_{y'^-} \rangle} \right)}. \tag{15}$$

E More experiments

E.1 Other backbone

Table 7 shows additional experiments on CLIP ViT-L/14, and the results demonstrate that CoOD is fully compatible with larger backbones. While most methods benefit from the stronger encoder, CoOD consistently maintains superior performance.

Table 7: OOD detection on ImageNet and CUB benchmarks with various OOD detectors and CLIP ViT-L/14. * indicates methods that require OOD labels (we use the ID-only version of Dual and ΔE to compare with CoOD).

Metr.	$\mathcal{D}$	MLS	KL	EBO	ViM	ODIN	MCM	Gap	GL/R	EOE*	NEG*	Dual	CSP*	COOK	ΔE	CoOD-F	CoOD	CoOD*
AUC	ImageNet	83.5	64.8	80.0	68.6	91.1	90.4	90.6	91.1	93.3	94.3	90.7	95.7	91.5	90.4	96.6	97.2	**97.8**
	CUB	80.5	64.6	80.1	69.8	77.5	77.6	70.8	76.7	79.1	80.1	78.8	81.1	77.6	76.2	82.2	85.3	**86.9**
FPR	ImageNet	48.6	80.5	51.2	63.7	37.4	37.8	37.2	39.3	27.7	25.3	37.2	23.1	34.4	38.3	14.3	13.0	**8.9**
	CUB	68.9	85.3	69.8	81.3	76.2	74.9	70.7	76.8	72.4	71.0	73.6	66.8	75.1	77.1	58.1	54.8	**53.6**

E.2 Additional benchmarks

Additional benchmarks include:

- **(i') ImageNet-OpenOOD**, while most existing methods follow the ImageNet-CLIP protocol [2], [3] evaluates under the OpenOOD benchmark. To ensure a fair and consistent comparison, we follow [3] and adopt the OpenOOD setting, which identifies near-OOD NINCO, ImageNet-O, and ImageNetOOD from ImageNet1K;
- **(i") ImageNet-54** uses 54 classes with clearly separable components to validate component-specific effectiveness of CoOD, whereas the OOD datasets are the same as in **(i)**;
- **(iv) Covariate-IDs on ObjectNet-104**, which uses the 104-class intersection of ImageNet and ObjectNet [62] to test the sensitivity of CCS to geometric variations (*e.g.*, viewpoint, rotation, and scale), as CCS relies on spatial structure. In this setting, ObjectNet serves as covariate-ID (*a.k.a.* close-set OOD [22]), while the real OOD (*a.k.a.* open-set OOD) sets are the same as in **(i)**;
- **(iv') Covariate-IDs on ObjectNet-54**, which is similar to **(iv)**, but the covariate-ID classes are the same as in **(i")**;
- **(iv") Covariate-IDs on ImageNet-R / -A / -Sketch / -V2**. Similar to *Setup-I* in [22], which distinguishes close-set OOD data from open-set OOD data on ImageNet-1K, we adopt the 314-class union of ImageNet-R and ImageNet-A as the covariate-shifted ID classes. The corresponding 314 classes in ImageNet-Sketch and ImageNet-V2 are collected together with ImageNet-R/-A and treated as close-set OOD data (*i.e.*, covariate-shifted but semantically overlapping), while the remaining 686 original ImageNet classes are used as open-set OOD data (semantic OODs). Note that this setting is more challenging, because all selected IDs are covariate-shifted and cover all corruption patterns (in contrast to the random 40%/60% split in original version of [22]), whereas OODs is purely semantic shifted.

E.3 Additional benchmarks for main experiments

We provide more experimental results on ImageNet1K-OpenOOD, ImageNet-54, and ImageNet-104 in Tables 8 and 9 for benchmarks **i', i"**, and **iv**. The results support the conclusions in the main paper.

Table 8: OOD detection on ImageNet-54 and ImageNet-104 with various OOD detectors and models. * indicates methods that require OOD labels [28] (we use the ID-only version of Dual and ΔE to compare with CoOD).

$\mathcal{D}$	Metr.	Tr.	MLS	KL	EBO	ViM	ODIN	MCM	Gap	GL/R	EOE*	NEG*	Dual	CSP*	COOK	ΔE	CoOD-F	CoOD	CoOD*
ImageNet-54	AUC	CLIP	81.5	93.1	81.5	84.5	92.8	93.4	90.8	93.6	90.3	96.9	93.7	97.7	93.4	93.9	96.9	98.4	**98.6**
		CoOp	90.2	91.6	90.2	87.2	93.3	93.9	92.1	94.0	93.2	95.3	94.1	96.5	93.9	94.1	95.1	98.7	**99.2**
		LoCoOp	83.7	91.2	83.7	85.1	93.3	95.5	94.2	95.6	92.6	91.4	95.7	93.5	95.5	95.6	96.0	98.6	**99.1**
		IDLike	87.4	93.1	87.4	86.3	93.2	92.7	93.3	93.0	91.9	93.6	95.2	95.3	92.8	95.1	94.5	98.6	**98.7**
		LoPro	81.2	91.7	81.2	84.4	92.9	93.2	90.7	93.4	89.8	93.7	93.5	96.8	93.2	93.6	96.2	97.8	**98.6**
	FPR_{95}	CLIP	46.1	25.1	46.1	31.3	23.6	23.9	29.6	23.6	30.2	14.2	23.0	10.1	23.8	22.9	13.9	9.7	**6.5**
		CoOp	31.1	30.9	31.1	27.1	24.6	23.8	27.3	23.4	24.3	18.4	22.8	14.8	23.7	23.3	22.2	6.4	**3.4**
		LoCoOp	43.9	31.9	43.9	29.2	24.3	18.4	21.9	17.8	23.6	28.5	17.4	22.8	18.3	18.1	17.2	6.0	**4.0**
		IDLike	39.7	23.9	39.7	28.8	23.6	27.3	23.1	26.9	29.2	25.0	18.1	19.5	27.2	18.6	22.6	6.8	**5.6**
		LoPro	46.1	28.1	46.2	31.7	23.7	24.3	29.3	23.9	30.6	21.9	23.6	14.0	24.2	23.1	15.7	9.7	**6.8**
ImageNet-104	AUC	CLIP	81.3	93.2	81.3	89.6	93.6	93.6	90.6	93.8	92.0	96.5	93.6	97.4	93.6	94.1	96.0	97.3	**98.3**
		CoOp	90.2	92.6	90.2	92.2	93.9	93.7	91.2	93.8	93.8	94.5	93.8	96.3	93.7	94.1	96.6	97.7	**98.6**
		LoCoOp	88.1	91.6	88.1	91.3	93.7	93.3	90.6	93.5	93.0	93.8	93.3	96.2	93.3	93.5	97.2	98.4	**98.9**
		IDLike	83.1	93.2	83.0	89.6	93.8	88.3	92.4	88.5	88.9	88.8	94.8	92.0	88.3	95.0	96.3	97.4	**98.2**
		LoPro	80.8	91.6	80.8	89.3	93.8	93.1	90.4	93.3	91.3	91.4	93.2	95.6	93.2	93.6	95.5	97.2	**98.4**
	FPR_{95}	CLIP	45.9	25.8	45.9	27.1	25.5	23.8	31.4	23.4	26.0	15.5	23.5	11.7	23.7	22.4	13.5	10.0	**7.4**
		CoOp	33.5	25.7	33.5	22.7	25.3	25.1	32.3	24.8	24.4	22.7	24.8	16.6	25.1	22.6	13.8	9.5	**6.3**
		LoCoOp	36.5	29.8	36.5	24.7	25.2	26.4	33.9	26.1	25.4	22.6	26.3	15.5	26.4	25.8	12.1	6.9	**4.7**
		IDLike	44.4	23.5	44.5	28.9	24.9	33.5	26.7	33.1	33.5	31.9	19.3	25.6	33.4	18.7	13.7	9.6	**7.5**
		LoPro	45.6	29.2	45.6	27.6	25.3	24.3	31.1	24.1	26.8	24.8	24.1	17.3	24.2	23.2	15.1	9.8	**7.5**

Table 9: OOD detection on ImageNet1K-OpenOOD with various OOD detectors and models. * indicates methods that require OOD labels.

Method	NINCO		ImageNet-O		ImageNetOOD		AVG	
	FPR_{95}	AUC	FPR_{95}	AUC	FPR_{95}	AUC	FPR_{95}	AUC
EBO	84.1	72.0	81.6	75.6	79.1	76.8	81.6	74.8
+TAG	83.1	71.1	79.7	77.1	78.3	78.0	80.4	75.4
MCM	79.6	73.6	75.9	79.5	81.0	78.3	78.8	77.1
+TAG	81.6	71.3	77.6	79.9	83.0	78.8	80.7	76.7
MLS	79.4	74.3	77.2	77.8	75.9	78.7	77.5	77.0
+TAG	77.7	73.9	74.5	79.5	75.2	80.1	75.8	77.8
GL	74.4	76.0	72.4	79.5	79.2	77.3	75.3	74.7
Gap	77.4	76.5	72.0	81.4	75.4	80.3	74.9	79.4
CoOD-F	75.6	74.4	54.2	82.2	58.6	81.8	62.8	79.4
CoOD	65.8	80.0	53.5	**83.4**	**51.4**	**84.6**	**56.9**	**82.7**
CoOD*	**61.4**	**81.8**	**52.6**	82.9	57.0	82.9	57.0	82.5
CoOp	81.5	70.2	75.4	79.8	80.4	78.6	79.1	76.2
+Gap	80.7	74.0	72.7	81.1	76.1	80.2	76.5	78.4
+CoOD-F	81.5	70.2	62.3	80.7	65.9	79.6	69.9	76.8
+CoOD	**52.2**	80.6	**52.6**	81.5	52.4	82.6	**52.4**	81.6
+CoOD*	69.3	81.1	53.9	83.0	52.3	83.1	58.5	**82.4**
ID-Like	81.2	71.8	82.5	68.4	88.5	62.4	84.1	67.5
+Gap	77.5	75.4	78.1	75.3	82.6	72.9	79.4	74.5
+CoOD-F	74.4	71.2	57.7	81.4	59.4	80.8	63.8	77.8
+CoOD	63.9	79.4	55.2	**82.6**	**52.2**	**83.9**	57.1	82.0
+CoOD*	60.0	**81.2**	58.0	81.6	56.0	82.6	58.0	81.8

E.4 Additional benchmarks for Covariate-ID

CoOD remains effective under challenging conditions. On the more challenging ObjectNet benchmarks **iv** and **iv'**, which introduce substantial covariate shifts (*e.g.* cluttered backgrounds, occlusions, extreme viewpoints, and scale variations) in Table 10, most methods experience noticeable performance degradation. In this setting, which stresses the impact of extreme geometric shift on CCS, CoOD maintains competitive performance, demonstrating robustness in complex and realistic settings.

Table 11 further confirms the effectiveness of our method under non-geometrical covariate shifts setting **iv"**. The strong performance of CoOD is likely due to CCS, which emphasizes compositional consistency and is less sensitive to appearance variations.

Table 10: OOD detection on ObjectNet-54 and ObjectNet-104 with various OOD detectors and models. * indicates methods that require OOD labels [28] (we use the ID-only version of Dual and ΔE to compare with CoOD).

$\mathcal{D}$	Metr.	Tr.	MLS	KL	EBO	ViM	ODIN	MCM	Gap	GL/R	EOE*	NEG*	Dual	CSP*	COOK	ΔE	CoOD-F	CoOD	CoOD*
ObjectNet-54	AUC	CLIP	86.8	90.9	86.8	84.9	91.7	88.4	83.5	88.7	90.6	96.8	88.6	98.0	88.4	89.5	91.0	94.3	**98.5**
		CoOp	87.8	92.2	87.8	85.5	89.8	87.8	82.6	88.1	90.4	92.8	88.2	94.8	87.8	89.1	91.9	95.3	**98.5**
		LoCoOp	88.0	88.3	87.9	85.7	90.4	90.7	87.5	91.0	92.5	93.5	90.9	95.4	90.7	91.7	92.4	95.4	**98.3**
		IDLike	94.4	93.5	94.4	90.8	90.3	96.0	87.5	96.2	95.8	97.2	92.2	98.0	96.1	93.0	93.9	97.8	**98.6**
		LoPro	86.5	90.8	86.5	84.8	91.8	88.7	83.5	88.9	90.5	95.8	88.9	98.0	88.7	89.9	91.0	94.5	**98.5**
	FPR_{95}	CLIP	39.1	31.7	39.1	40.3	27.5	38.4	45.5	38.1	34.2	16.8	37.5	9.4	38.4	35.8	22.7	23.2	**8.0**
		CoOp	41.0	31.8	41.0	40.2	33.4	41.3	47.5	40.9	36.3	29.9	40.2	24.2	41.2	37.8	21.4	22.0	**7.9**
		LoCoOp	38.8	39.8	38.8	37.4	31.8	33.3	39.0	32.6	27.3	25.0	32.4	18.4	33.3	30.0	26.1	20.6	**8.0**
		IDLike	19.1	24.6	19.0	25.6	30.8	16.8	38.8	16.6	17.1	10.8	30.0	8.4	16.8	27.9	20.2	10.0	**7.0**
		LoPro	39.1	31.7	39.1	40.3	27.5	38.3	45.2	37.8	33.7	50.9	37.5	8.9	38.2	35.0	31.0	22.9	**8.0**
ObjectNet-104	AUC	CLIP	92.8	91.0	92.8	89.6	95.1	89.3	84.2	89.6	93.3	98.4	89.4	99.0	89.2	90.4	92.0	93.6	**99.2**
		CoOp	95.9	92.8	95.9	92.7	93.7	88.7	82.8	89.0	94.4	97.6	88.9	98.2	88.7	90.2	92.4	94.2	**98.7**
		LoCoOp	89.2	91.0	89.2	88.3	93.5	87.8	81.7	88.1	91.3	93.5	87.9	96.4	87.8	89.0	94.6	95.1	**98.7**
		IDLike	89.5	94.4	89.5	88.3	94.5	95.8	85.5	95.9	96.8	93.6	91.8	95.9	95.8	92.7	96.2	97.4	**98.9**
		LoPro	91.9	90.4	91.9	89.2	95.1	88.9	83.5	89.1	92.7	97.7	89.0	99.1	88.8	90.1	91.9	93.5	**99.3**
	FPR_{95}	CLIP	26.0	36.3	26.0	32.1	26.1	40.0	49.5	39.3	28.6	8.1	39.7	4.0	40.0	36.6	26.6	23.9	**3.5**
		CoOp	19.7	29.7	19.7	27.3	29.5	44.8	56.1	44.5	27.6	13.7	44.5	9.7	45.0	39.2	27.3	24.2	**7.2**
		LoCoOp	39.2	37.0	39.3	36.7	29.6	46.3	57.7	45.8	35.9	27.0	46.2	18.1	46.3	43.3	21.1	20.9	**7.3**
		IDLike	36.5	23.4	36.5	36.5	27.2	19.2	47.3	19.1	15.9	26.0	33.4	20.2	19.3	29.7	17.6	12.0	**5.8**
		LoPro	26.9	36.9	26.9	32.3	26.2	40.1	49.3	39.7	28.6	10.8	40.0	4.0	40.2	36.7	26.8	24.0	**3.1**

Table 11: Additional benchmark on ImageNet-A/-R/-S/-V2 covariate IDs *vs.* original ImageNet semantic OODs with various detectors. * indicates methods that require OOD labels (we use the ID-only version of Dual and ΔE to compare with CoOD).

Metric	MLS	KL	EBO	ViM	ODIN	MCM	Gap	GL	EOE*	NEG*	Dual	CSP*	COOK	ΔE	CoOD-F	CoOD	CoOD*
AUC	61.3	49.8	59.4	64.7	58.0	66.3	67.6	69.0	67.0	68.5	65.7	65.0	50.8	63.6	68.3	79.6	**80.1**
FPR	87.1	97.9	87.8	80.1	89.3	80.0	78.9	79.3	80.7	82.5	82.2	84.2	90.4	84.9	80.2	58.1	**55.8**

E.5 Significance tests

We perform significance tests via paired t-test over 5 random seeds. For each main experiment and claim, **ALL** performance improvements that "CoOD *vs.* SOTA method w/o OOD label" and "CoOD* *vs.* SOTA method $w/$ OOD label" under **(i) standard OOD protocol**, **(ii) fine-grained OOD**, and **(iii) compositional OOD** are statistically significant ($p < 0.001$).

E.6 Failure cases

Figure 7 illustrates two representative segmentation failures, complementing the successful foreground and component localization shown in the main text. **Top:** When the car lacks a spoiler, the component localization activates a prior region near the front of the vehicle, suppressing the response of the car body and resulting in inaccurate segmentation. Nevertheless, soft suppression preserves partial ID evidence, allowing the model to retain some component score. **Bottom:** When the initial foreground localization fails, subsequent component segmentation cannot recover the relevant regions. Since zoom-in involves hard cropping, the missing foreground evidence cannot be recovered, leading to an OOD prediction. These cases highlight the different impacts of imperfect component masks and foreground localization failures.

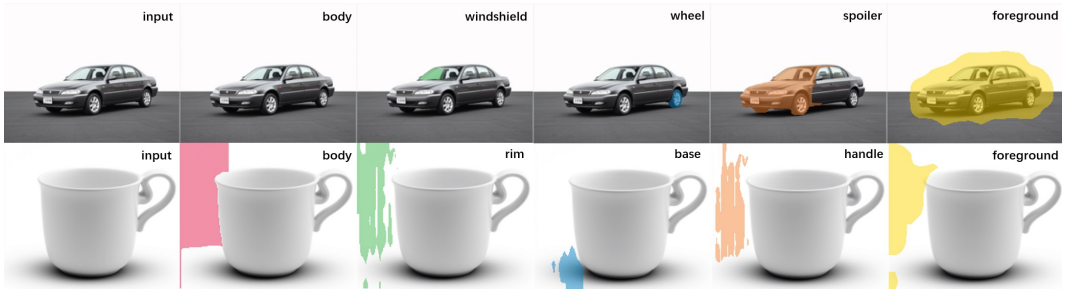

Figure 7: Representative failure cases of component localization.

F Semantic text selection and component text generation

F.1 Component text generation

We generate component text by prompting ChatGPT-4o-2024-11-20:

From a 360-degree perspective, examine each of the following categories: ‘[$'class_1'$,$'class_2'$,...]‘.

For each item:

1. **Identify a compact, non-overlapping set of visually distinct parts or components** that are clearly visible from multiple angles and the visual size is large enough ("flange" is too small for "snorkel"). Focus on features that help **discriminate the object from visually similar categories** using CLIP-style segmentation or recognition.

2. **Build a part taxonomy tree** per object, from general to specific. Select final **leaf-level parts** that are:

* Highly discriminative

* Commonly visible

* Not too fine-grained or rare

* If necessary, back off to their parent or mid-level nodes for visual reliability.

* If necessary, you can make an analogy for classes that hard to discompose into parts, e.g., chain mail can use parts of clothing, such as: "sleeves", "hem", etc.

3. **Clean the part names** by removing or summarizing:

* Descriptors (e.g. "small," "left," "top") * Redundant modifiers * Directional/locational terms * Duplicate terms

4. **Avoid excessive part lists**:

* **4 parts per object** is preferred. * **More than 4, less than 6** is allowed if all are visually salient. * **Fewer than 4** is conditionally acceptable **only if justified**, e.g., the object is visually minimal or non-decomposable (e.g. "golf ball").

5. Use **natural, social-media-friendly English terms** for parts (e.g. "handle" not "manubrium").

* If two or more part names refer to nearly the same or visually proximate structure, combine them using the format: **"limb-paw"**. Only merge parts that are visually indistinct or spatially inseparable.

Avoid generating overly technical or obscure terms. Keep focus on what is **recognizable by a non-expert viewer**.

Avoid phrases if a single word is clear and discriminative.

6. The results should be in the format of python dict, e.g. for class name of "cat", we have caption = {"cat": ["head", "limb-paw", "tail", "trunk"],}

F.2 Semantic classes and the generated component text

Class names and the corresponding component names are as follows:

ImageNet

Class/Component names: PART_LABEL = { "snorkel": ["mouthpiece", "tube", "glasses", "face"], "syringe": ["barrel", "plunger", "needle

hub", "needle"], "binoculars": ["objective lens", "eyepiece", "focus knob", "bridge"], "cannon": ["barrel", "wheels", "carriage", "breech"], "crane_machine": ["boom", "counterweight", "cab", "hoist", "mast"], "desktop_computer": ["tower case", "power button", "ports panel", "vent"], "gas_pump": ["nozzle", "hose", "display", "handle"], "muzzle": ["strap", "cover", "head", "snout"], "seat_belt": ["webbing", "buckle", "seat", "human"], "patio": ["paving", "furniture", "canopy", "plant"], "home_theater": ["screen", "speakers", "table", "couch"], "totem_pole": ["base", "carved figures", "top figure"], "grille": ["body", "light", "license plate", "emblem"], "mountain_tent": ["canopy", "entrance", "rainfly", "guylines"], "scoreboard": ["display panel", "frame", "supports", "digits"], "cocktail_shaker": ["tin", "strainer", "cap", "body"], "kimono": ["sleeves", "collar", "obi"], "whiskey_jug": ["body", "neck", "handle", "stopper"], "knee_pad": ["pad", "straps", "leg"], "book_jacket": ["cover", "pages", "title", "spine"], "crash_helmet": ["shell", "visor", "chin strap", "padding"], "vestment": ["chasuble", "stole", "cuffs"], "cloak": ["hood", "body", "hem"], "scabbard": ["sheath", "belt loop", "mouth", "handle-grip", "blade-knife"], "beer_glass": ["rim", "body", "base"], "swab": ["shaft", "tip-head", "handle", "strand"], "drilling_platform": ["sea", "tower", "supports-legs", "deck"], "pencil_box": ["lid", "body", "hinge", "zipper"], "punching_bag": ["bag body", "chain strap", "mount"], "pencil_sharpener": ["body", "handle", "blade", "shavings receptacle"], "shower_cap": ["cap body", "head", "hair", "elastic band"], "trolleybus": ["pantograph", "body", "wheels", "windows"], "perfume": ["bottle", "spray nozzle", "cap"], "crate": ["slats", "frame"], "ballpoint": ["tip", "barrel", "cap"], "comic_book": ["cover", "pages", "title", "spine"], "wooden_spoon": ["bowl", "handle"], "ice_lolly": ["stick", "frozen block", "head", "hand"], "carbonara": ["spaghetti", "sauce", "marinara", "bolognese", "plate"], "caldron": ["body", "handle", "lid", "legs"], "backpack": ["compartment", "pocket", "strap", "handle"], "banana": ["peel", "flesh", "stem", "tip"], "Band_Aid": ["pad", "adhesive", "backing"], "shopping_basket": ["body", "handle", "rim", "base", "mesh"], "bath_towel": ["body", "hem", "hanger loop", "border"], "beer_bottle": ["neck", "body", "label", "cap"], "park_bench": ["seat", "backrest", "armrest", "legs"], "binder": ["cover", "spine", "rings", "label"], "bottlecap": ["top", "side wall", "liner", "bottle"], "French_loaf": ["crust", "scored-slashes", "tip", "slice-face", "wrapper", "basket"], "broom": ["handle", "brush-bristles", "ferrule", "shaft", "tip-head", "strand"], "bucket": ["body", "handle", "rim", "base"], "cleaver": ["blade", "table", "board", "handle"], "can_opener": ["handle", "cutting-wheel", "gear", "arm", "can"], "candle": ["body", "wick", "drip", "base"], "cellular_telephone": ["screen", "panel", "camera", "buttons", "speaker"], "hamper": ["basket-body", "lid", "handles", "base"], "espresso_maker": ["body", "group-head", "portafilter", "steam-wand", "water-tank"], "combination_lock": ["dial", "body", "shackle", "number ring"], "mouse": ["buttons", "scroll-wheel", "body", "wire"], "table_lamp": ["base", "stem", "lampshade", "bulb socket", "wire"], "dishrag": ["dish", "body", "hem", "hanger-loop"], "doormat": ["door", "surface", "edge", "backing"], "Loafer": ["upper", "sole", "heel", "vamp"], "power_drill": ["chuck", "body", "trigger", "handle", "battery pack", "wire"], "cup": ["rim", "body", "handle", "base"], "plate_rack": ["slots", "frame", "base", "plate"] , "envelope": ["body", "flap", "seal area", "window"], "electric_fan": ["blades", "grill", "motor", "base"], "frying_pan": ["body", "handle", "base", "rim"], "gown": ["bodice", "skirt", "sleeves", "neckline"], "hand_blower": ["nozzle", "barrel", "handle", "air-intake", "wire"], "hammer": ["head", "claw", "face", "handle"], "iron": ["soleplate", "tank", "handle", "control panel"], "jean": ["waistband", "legs", "pockets", "fly"], "computer_keyboard": ["keycaps", "frame", "spacebar", "function-row",

"numpad"], "ladle": ["bowl", "handle", "hook"], "lampshade": ["shade",
"rim", "fitting", "lining"], "laptop": ["screen", "keyboard", "touchpad",
"hinge", "base"], "lemon": ["peel", "pulp", "stem-end"], "letter_opener":
["blade", "handle", "point-tip"], "lighter": ["body", "nozzle", "wheel
igniter", "fuel window"], "lipstick": ["cap", "tube", "bullet", "base"],
"matchstick": ["head", "shaft", "box"], "measuring_cup": ["body",
"handle", "spout", "markings", "table"], "microwave": ["door", "control
panel", "turntable", "cavity"], "mixing_bowl": ["bowl", "rim", "base"],
"monitor": ["screen", "bezel", "stand", "port", "wire"], "coffee_mug":
["handle", "rim", "body", "base", "plate"], "nail": ["head", "shank-body",
"tip-point"], "necklace": ["chain", "pendant", "clasp", "neck"], "orange":
["peel", "segments", "stem-end", "seed"], "padlock": ["body", "shackle",
"keyway"], "paintbrush": ["bristles", "ferrule", "handle"], "paper_towel":
["sheet", "perforation", "pack"], "pill_bottle": ["cap", "body",
"label", "rim"], "pillow": ["shell", "filling", "seam", "bed", "head"],
"pitcher": ["spout", "handle", "body", "base"], "plastic_bag": ["body",
"handles", "gusset"], "plate": ["rim", "center", "underside"], "plunger":
["handle", "rubber cup", "shaft"], "pop_bottle": ["neck", "cap", "body",
"label", "base"], "space_heater": ["grill", "control panel", "housing",
"base"], "printer": ["paper-tray", "output-tray", "control-panel",
"cartridge"], "remote_control": ["buttons", "directional-pad", "battery
compartment", "infrared emitter"], "rule": ["edge", "markings", "body"],
"running_shoe": ["upper", "sole", "tongue", "laces"], "safety_pin":
["pin", "clasp", "spring"], "saltshaker": ["body", "cap", "base", "holes"],
"sandal": ["sole", "straps", "buckle", "footbed"], "screw": ["head",
"shank", "thread", "point"], "shovel": ["blade", "shaft", "grip",
"collar"], "sleeping_bag": ["shell", "zipper", "hood", "insulation"],
"soap_dispenser": ["pump", "bottle", "nozzle", "base"], "sock": ["cuff",
"arch", "toe", "heel"], "soup_bowl": ["rim", "bowl", "base"], "spatula":
["blade", "handle", "neck"], "loudspeaker": ["cone", "grille", "cabinet",
"mount", "magnet"], "strainer": ["mesh", "rim", "handle", "hook"], "teddy":
["head", "limbs", "body", "face"], "suit": ["jacket", "trousers", "lapel",
"buttons"], "sunglasses": ["lenses", "frame", "temples", "bridge"],
"sweatshirt": ["hood", "pocket", "cuffs", "hem"], "swimming_trunks":
["waistband", "leg openings", "pocket", "drawstring"], "jersey": ["pocket",
"sleeves", "collar", "logo-number", "hem"], "television": ["screen",
"bezel", "stand", "back panel"], "teapot": ["body", "spout", "handle",
"lid"], "racket": ["head", "stringbed", "frame", "handle"], "toaster":
["slots", "lever", "body", "tray", "grille", "knobs"], "toilet_tissue":
["tube", "paper", "perforation"], "ashcan": ["lid", "body", "rim",
"foot pedal"], "tray": ["flat surface", "rim", "handles"], "umbrella":
["canopy", "ribs", "shaft", "handle"], "vacuum": ["hose", "housing",
"brush-head", "handle"], "vase": ["neck", "body", "base", "rim"], "wallet":
["compartment", "card-slots", "coin-pocket", "fold"], "digital_watch":
["face", "strap", "buttons", "buckle"], "water_bottle": ["cap", "neck",
"body", "base"], "dumbbell": ["handle", "weight-plates", "collar"],
"scale": ["platform", "display", "buttons", "base"], "whistle": ["body",
"mouthpiece", "pea", "ring"], "wine_bottle": ["neck", "body", "closure",
"label"], "mitten": ["palm", "thumb", "cuff", "insulation"], "wok":
["bowl", "handle", "base"], "tench": ["head", "dorsal fin", "tail",
"pectoral fins", "gills"], "goldfish": ["head", "dorsal fin", "tail",
"pectoral fins", "eyes"], "great_white_shark": ["snout", "dorsal fin",
"tail", "pectoral fins", "gill slits", "teeth"], "tiger_shark": ["snout",
"dorsal fin", "tail", "pectoral fins", "gill slits", "teeth"], "hammerhead":
["hammer head", "dorsal fin", "tail", "pectoral fins", "eyes", "teeth"],
"electric_ray": ["pectoral disc", "tail", "eyes", "mouth", "pectoral
fin"], "stingray": ["pectoral disc", "tail barb", "eyes", "mouth",
"pectoral fin"], "cock": ["comb", "wattle", "beak", "wings", "tail",
"legs"], "hen": ["comb", "wattle", "beak", "wings", "tail", "legs"],

"ostrich": ["head", "neck", "wings", "legs"], "brambling": ["beak",
"head", "wings", "tail"], "goldfinch": ["beak", "head", "wings", "tail"],
"house_finch": ["beak", "head", "wings", "tail"], "junco": ["beak",
"head", "wings", "tail"], "indigo_bunting": ["beak", "head", "wings",
"tail"], "robin": ["beak", "head", "wings", "tail"], "bulbul": ["beak",
"head", "crest", "wings", "tail"], "jay_bird": ["beak", "head", "crest",
"wings", "tail"], "crane_bird": ["beak", "head", "crest", "wings",
"tail"], "magpie": ["beak", "head", "long tail", "wings"], "chickadee":
["beak", "head", "wings", "tail"], "water_ouzel": ["beak", "head",
"wings", "tail"], "kite": ["beak", "head", "wings", "talons", "tail"],
"bald_eagle": ["beak", "head", "wings", "talons", "tail"], "vulture":
["beak", "head", "neck", "wings", "legs"], "great_grey_owl": ["face
disc", "beak", "wings", "talons", "tail"], "European_fire_salamander":
["head", "limbs", "tail", "eyes"], "common_newt": ["head", "limbs", "tail",
"eyes"], "eft": ["head", "limbs", "tail", "eyes"], "spotted_salamander":
["head", "limbs", "tail", "spots"], "axolotl": ["head", "gills", "limbs",
"tail"], "bullfrog": ["head", "eyes", "hind legs", "mouth"], "tree_frog":
["head", "toe pads", "hind legs", "eyes"], "tailed_frog": ["head",
"tail organ", "hind legs", "eyes"], "loggerhead": ["head", "shell",
"flippers", "tail", "belly"], "leatherback_turtle": ["head", "leathery
shell", "flippers", "tail", "belly"], "mud_turtle": ["head", "shell",
"legs", "tail", "belly"], "terrapin": ["head", "shell", "legs", "tail",
"belly"], "box_turtle": ["head", "domed shell", "legs", "tail", "belly"],
"banded_gecko": ["head", "legs", "tail", "eyes"], "common_iguana": ["head
crest", "dorsal crest", "legs", "tail", "dewlap"], "American_chameleon":
["head helmet", "prehensile tail", "feet", "eyes"], "whiptail": ["head",
"legs", "tail"], "agama": ["head", "dorsal crest", "legs", "tail"],
"frilled_lizard": ["head", "frill", "legs", "tail"], "alligator_lizard":
["head", "legs", "tail"], "Gila_monster": ["head", "legs", "tail"],
"green_lizard": ["head", "legs", "tail"], "African_chameleon": ["head
helmet", "prehensile tail", "feet", "eyes"], "Komodo_dragon": ["head",
"neck", "legs", "tail"], "African_crocodile": ["snout", "eyes and
nostrils", "legs", "tail"], "American_alligator": ["snout", "eyes and
nostrils", "legs", "tail"], "triceratops": ["head frill", "horns",
"beak", "legs"], "thunder_snake": ["head", "body", "belly", "tail"],
"ringneck_snake": ["head", "neck ring", "body", "tail"], "hognose_snake":
["upturned snout", "head", "body pattern", "tail"], "green_snake":
["head", "slender body", "tail"], "king_snake": ["head", "banded
pattern", "body", "tail"], "garter_snake": ["head", "striped body",
"tail"], "water_snake": ["head", "body pattern", "tail"], "vine_snake":
["head", "very slender body", "eyes", "tail"], "night_snake": ["head",
"body pattern", "tail"], "boa_constrictor": ["head", "muscular body",
"tail"], "rock_python": ["head", "heavy body", "tail"], "Indian_cobra":
["head", "hood", "fangs", "tail"], "green_mamba": ["head", "slender
body", "fangs", "tail"], "sea_snake": ["head", "flattened tail",
"vent", "tail tip"], "horned_viper": ["head", "horn", "fangs", "tail"],
"diamondback": ["head", "rattle", "fangs", "tail"], "sidewinder": ["head",
"rattle", "belly scales", "tail"], "trilobite": ["head shield", "thoracic
segments", "tail shield", "compound eyes"], "harvestman": ["body", "legs",
"mouthparts"], "scorpion": ["pincers", "body", "tail", "legs", "stinger"],
"black_and_gold_garden_spider": ["head", "abdomen", "legs", "spinnerets"],
"barn_spider": ["head", "abdomen", "legs", "spinnerets"], "garden_spider":
["head", "abdomen", "legs", "spinnerets"], "black_widow": ["head",
"abdomen", "legs", "spinnerets"], "tarantula": ["head", "abdomen", "thick
legs", "pedipalps"], "wolf_spider": ["head", "abdomen", "legs", "eye
cluster"], "tick": ["body", "mouthparts", "legs"], "centipede": ["head",
"body segments", "many legs", "claws"], "black_grouse": ["beak", "head
comb", "wings", "tail"], "ptarmigan": ["beak", "head", "wings", "tail"],
"ruffed_grouse": ["beak", "head", "ruff", "wings"], "prairie_chicken":

```
["beak", "air sacs", "wings", "tail"], "peacock": ["head", "crest",
"display tail", "wings"], "quail": ["beak", "head crest", "wings", "tail"],
"partridge": ["beak", "head", "wings", "tail"], "African_grey": ["beak",
"head", "wings", "tail"], "macaw": ["beak", "head", "wings", "long
tail", "feet"], "sulphur-crested_cockatoo": ["crest", "beak", "head",
"wings", "tail"], "lorikeet_parrot": ["beak", "head", "wings", "tail"],
"coucal_bird": ["beak", "head", "long tail", "wings"], "bee_eater":
["long beak", "head", "wings", "tail"], "hornbill": ["beak", "head",
"wings", "tail", "legs"], "hummingbird": ["beak", "head", "wings",
"tail"], "jacamar": ["beak", "head", "wings", "tail"], "toucan": ["beak",
"head", "wings", "tail"], "drake": ["beak", "head", "wings", "tail",
"feet"], "red-breasted_merganser": ["beak", "head", "wings", "tail",
"feet"], "goose": ["beak", "head", "wings", "tail", "feet"], "black_swan":
["bill", "head", "neck", "wings", "feet"], "tusker": ["head", "tusks",
"trunk", "ears", "legs"], "echidna": ["snout", "spines", "legs",
"eyes"], "platypus": ["bill", "webbed feet", "tail", "legs"], "wallaby":
["head", "ears", "forepaws", "hind legs", "tail"], "koala": ["head",
"ears", "forepaws", "hind legs", "tail"], "wombat": ["head", "ears",
"forepaws", "hind legs", "tail"], "jellyfish": ["bell", "tentacles",
"mouth"], "sea_anemone": ["base", "tentacles", "mouth"], "brain_coral":
["lobes", "ridges", "valleys"], "flatworm": ["head", "tail", "body"],
"nematode": ["head", "tail", "body"], "conch": ["shell", "aperture",
"foot", "tentacles"], "snail": ["shell", "aperture", "foot", "tentacles"],
"land_slug": ["head", "mantle", "foot", "tentacles"], "sea_slug": ["head",
"mantle", "gills", "foot"], "chiton": ["shell plates", "girdle", "foot"],
"chambered_nautilus": ["shell", "tentacles", "hood"], "Dungeness_crab":
["shell", "claws", "walking legs", "antennae"], "rock_crab": ["shell",
"claws", "walking legs", "antennae"], "fiddler_crab": ["shell", "major
claw", "walking legs", "eye stalks"], "king_crab": ["shell", "large
legs", "claws", "antennae"], "American_lobster": ["shell", "claws",
"tail fan", "walking legs", "antennae"], "spiny_lobster": ["shell",
"antennae", "tail fan", "walking legs"], "crayfish": ["shell", "claws",
"tail fan", "walking legs", "antennae"], "hermit_crab": ["shell",
"claws", "walking legs", "antennae"], "isopod": ["body segments", "legs",
"antennae"], "white_stork": ["beak", "head", "neck", "wings", "legs"],
"black_stork": ["beak", "head", "neck", "wings", "legs"], "spoonbill":
["bill", "head", "neck", "wings", "legs"], "flamingo": ["bill", "head",
"neck", "wings", "legs"], "little_blue_heron": ["beak", "head", "neck",
"wings", "legs"], "American_egret": ["beak", "head", "neck", "wings",
"legs"], "bittern": ["beak", "head", "neck", "wings", "legs"], "limpkin":
["beak", "head", "neck", "wings", "legs"], "European_gallinule": ["beak",
"head", "wings", "legs", "tail"], "American_coot": ["bill", "head",
"wings", "legs", "feet"], "bustard": ["beak", "head", "wings", "legs",
"tail"], "ruddy_turnstone": ["beak", "head", "wings", "legs", "tail"],
"red-backed_sandpiper": ["beak", "head", "wings", "legs", "tail"],
"redshank": ["beak", "head", "wings", "legs", "tail"], "dowitcher":
["beak", "head", "wings", "legs", "tail"], "oystercatcher": ["beak",
"head", "wings", "legs", "tail"], "pelican": ["beak", "head", "throat
pouch", "wings", "feet"], "king_penguin": ["beak", "head", "flippers",
"feet", "tail"], "albatross": ["beak", "head", "wings", "tail", "legs"],
"grey_whale": ["head", "blowhole", "flippers", "tail flukes", "dorsal
ridge"], "killer_whale": ["head", "dorsal fin", "flippers", "tail flukes"],
"dugong": ["head", "flippers", "tail flukes"], "sea_lion": ["head",
"flippers", "whiskers", "tail"], "Chihuahua": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Japanese_spaniel": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Maltese_dog": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Pekinese": ["head", "ears", "muzzle", "body",
"legs", "tail"], "Shih-Tzu": ["head", "ears", "muzzle", "body", "legs",
"tail"], "Blenheim_spaniel": ["head", "ears", "muzzle", "body", "legs",
```

"tail"], "papillon": ["head", "ears", "muzzle", "body", "legs", "tail"],
"toy_terrier": ["head", "ears", "muzzle", "body", "legs", "tail"],
"Rhodesian_ridgeback": ["head", "ears", "muzzle", "body", "legs", "tail"],
"Afghan_hound": ["head", "ears", "muzzle", "body", "legs", "tail"],
"basset": ["head", "ears", "muzzle", "body", "legs", "tail"], "beagle":
["head", "ears", "muzzle", "body", "legs", "tail"], "bloodhound": ["head",
"ears", "muzzle", "body", "legs", "tail"], "bluetick": ["head", "ears",
"muzzle", "body", "legs", "tail"], "black-and-tan_coonhound": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Walker_hound": ["head",
"ears", "muzzle", "body", "legs", "tail"], "English_foxhound": ["head",
"ears", "muzzle", "body", "legs", "tail"], "redbone": ["head", "ears",
"muzzle", "body", "legs", "tail"], "borzoi": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Irish_wolfhound": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Italian_greyhound": ["head", "ears", "muzzle",
"body", "legs", "tail"], "whippet": ["head", "ears", "muzzle", "body",
"legs", "tail"], "Ibizan_hound": ["head", "ears", "muzzle", "body",
"legs", "tail"], "Norwegian_elkhound": ["head", "ears", "muzzle", "body",
"legs", "tail"], "otterhound": ["head", "ears", "muzzle", "body", "legs",
"tail"], "Saluki": ["head", "ears", "muzzle", "body", "legs", "tail"],
"Scottish_deerhound": ["head", "ears", "muzzle", "body", "legs", "tail"],
"Weimaraner": ["head", "ears", "muzzle", "body", "legs", "tail"],
"Staffordshire_bullterrier": ["head", "ears", "muzzle", "body", "legs",
"tail"], "American_Staffordshire_terrier": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Bedlington_terrier": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Border_terrier": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Kerry_blue_terrier": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Irish_terrier": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Norfolk_terrier": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Norwich_terrier": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Yorkshire_terrier": ["head", "ears", "muzzle",
"body", "legs", "tail"], "wire-haired_fox_terrier": ["head", "ears",
"muzzle", "body", "legs", "tail"], "Lakeland_terrier": ["head", "ears",
"muzzle", "body", "legs", "tail"], "Sealyham_terrier": ["head", "ears",
"muzzle", "body", "legs", "tail"], "Airedale": ["head", "ears", "muzzle",
"body", "legs", "tail"], "cairn": ["head", "ears", "muzzle", "body",
"legs", "tail"], "Australian_terrier": ["head", "ears", "muzzle", "body",
"legs", "tail"], "Dandie_Dinmont": ["head", "ears", "muzzle", "body",
"legs", "tail"], "Boston_bull": ["head", "ears", "muzzle", "body", "legs",
"tail"], "miniature_schnauzer": ["head", "ears", "muzzle", "body", "legs",
"tail"], "giant_schnauzer": ["head", "ears", "muzzle", "body", "legs",
"tail"], "standard_schnauzer": ["head", "ears", "muzzle", "body", "legs",
"tail"], "Scotch_terrier": ["head", "ears", "muzzle", "body", "legs",
"tail"], "Tibetan_terrier": ["head", "ears", "muzzle", "body", "legs",
"tail"], "silky_terrier": ["head", "ears", "muzzle", "body", "legs",
"tail"], "soft-coated_wheaten_terrier": ["head", "ears", "muzzle", "body",
"legs", "tail"], "West_Highland_white_terrier": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Lhasa": ["head", "ears", "muzzle", "body",
"legs", "tail"], "flat-coated_retriever": ["head", "ears", "muzzle",
"body", "legs", "tail"], "curly-coated_retriever": ["head", "ears",
"muzzle", "body", "legs", "tail"], "golden_retriever": ["head", "ears",
"muzzle", "body", "legs", "tail"], "Labrador_retriever": ["head", "ears",
"muzzle", "body", "legs", "tail"], "Chesapeake_Bay_retriever": ["head",
"ears", "muzzle", "body", "legs", "tail"], "German_short-haired_pointer":
["head", "ears", "muzzle", "body", "legs", "tail"], "vizsla": ["head",
"ears", "muzzle", "body", "legs", "tail"], "English_setter": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Irish_setter": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Gordon_setter": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Brittany_spaniel": ["head",
"ears", "muzzle", "body", "legs", "tail"], "clumber": ["head", "ears",

"muzzle", "body", "legs", "tail"], "English_springer": ["head", "ears",
"muzzle", "body", "legs", "tail"], "Welsh_springer_spaniel": ["head",
"ears", "muzzle", "body", "legs", "tail"], "cocker_spaniel": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Sussex_spaniel": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Irish_water_spaniel": ["head",
"ears", "muzzle", "body", "legs", "tail"], "kuvasz": ["head", "ears",
"muzzle", "body", "legs", "tail"], "schipperke": ["head", "ears", "muzzle",
"body", "legs", "tail"], "groenendael": ["head", "ears", "muzzle", "body",
"legs", "tail"], "malinois": ["head", "ears", "muzzle", "body", "legs",
"tail"], "briard": ["head", "ears", "muzzle", "body", "legs", "tail"],
"kelpie": ["head", "ears", "muzzle", "body", "legs", "tail"], "komondor":
["head", "ears", "muzzle", "body", "legs", "tail"], "Old_English_sheepdog":
["head", "ears", "muzzle", "body", "legs", "tail"], "Shetland_sheepdog":
["head", "ears", "muzzle", "body", "legs", "tail"], "collie": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Border_collie": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Bouvier_des_Flandres": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Rottweiler": ["head", "ears",
"muzzle", "body", "legs", "tail"], "German_shepherd": ["head", "ears",
"muzzle", "body", "legs", "tail"], "Doberman": ["head", "ears", "muzzle",
"body", "legs", "tail"], "miniature_pinscher": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Greater_Swiss_Mountain_dog": ["head", "ears",
"muzzle", "body", "legs", "tail"], "Bernese_mountain_dog": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Appenzeller": ["head",
"ears", "muzzle", "body", "legs", "tail"], "EntleBucher": ["head",
"ears", "muzzle", "body", "legs", "tail"], "boxer": ["head", "ears",
"muzzle", "body", "legs", "tail"], "bull_mastiff": ["head", "ears",
"muzzle", "body", "legs", "tail"], "Tibetan_mastiff": ["head", "ears",
"muzzle", "body", "legs", "tail"], "French_bulldog": ["head", "ears",
"muzzle", "body", "legs", "tail"], "Great_Dane": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Saint_Bernard": ["head", "ears", "muzzle",
"body", "legs", "tail"], "Eskimo_dog": ["head", "ears", "muzzle", "body",
"legs", "tail"], "malamute": ["head", "ears", "muzzle", "body", "legs",
"tail"], "Siberian_husky": ["head", "ears", "muzzle", "body", "legs",
"tail"], "dalmatian": ["head", "ears", "muzzle", "body", "legs", "tail"],
"affenpinscher": ["head", "ears", "muzzle", "body", "legs", "tail"],
"basenji": ["head", "ears", "muzzle", "body", "legs", "tail"], "pug":
["head", "ears", "muzzle", "body", "legs", "tail"], "Leonberg": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Newfoundland": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Great_Pyrenees": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Samoyed": ["head", "ears",
"muzzle", "body", "legs", "tail"], "Pomeranian": ["head", "ears", "muzzle",
"body", "legs", "tail"], "chow": ["head", "ears", "muzzle", "body", "legs",
"tail"], "keeshond": ["head", "ears", "muzzle", "body", "legs", "tail"],
"Brabancon_griffon": ["head", "ears", "muzzle", "body", "legs", "tail"],
"Pembroke": ["head", "ears", "muzzle", "body", "legs", "tail"], "Cardigan":
["head", "ears", "muzzle", "body", "legs", "tail"], "toy_poodle": ["head",
"ears", "muzzle", "body", "legs", "tail"], "miniature_poodle": ["head",
"ears", "muzzle", "body", "legs", "tail"], "standard_poodle": ["head",
"ears", "muzzle", "body", "legs", "tail"], "Mexican_hairless": ["head",
"ears", "muzzle", "body", "legs", "tail"], "timber_wolf": ["head",
"ears", "muzzle", "body", "legs", "tail"], "white_wolf": ["head",
"ears", "muzzle", "body", "legs", "tail"], "red_wolf": ["head", "ears",
"muzzle", "body", "legs", "tail"], "coyote": ["head", "ears", "muzzle",
"body", "legs", "tail"], "dingo": ["head", "ears", "muzzle", "body",
"legs", "tail"], "dhole_dog": ["head", "ears", "muzzle", "body", "legs",
"tail"], "African_hunting_dog": ["head", "ears", "muzzle", "body", "legs",
"tail"], "hyena": ["head", "ears", "muzzle", "body", "legs", "tail"],
"red_fox": ["head", "ears", "muzzle", "body", "legs", "tail"], "kit_fox":
["head", "ears", "muzzle", "body", "legs", "tail"], "Arctic_fox": ["head",

"ears", "muzzle", "body", "legs", "tail"], "grey_fox": ["head", "ears", "muzzle", "body", "legs", "tail"], "tabby_cat": ["head", "ears", "muzzle", "body", "legs", "tail"], "tiger_cat": ["head", "ears", "muzzle", "body", "legs", "tail"], "Persian_cat": ["head", "ears", "eyes", "body", "tail"], "Siamese_cat": ["head", "ears", "eyes", "body", "tail"], "Egyptian_cat": ["head", "ears", "eyes", "body", "tail"], "cougar": ["head", "ears", "eyes", "body", "tail"], "lynx": ["head", "ears", "eyes", "body", "tail"], "leopard": ["head", "ears", "eyes", "body", "tail"], "snow_leopard": ["head", "ears", "eyes", "body", "tail"], "jaguar": ["head", "ears", "eyes", "body", "tail"], "lion": ["head", "mane", "eyes", "body", "tail"], "tiger": ["head", "ears", "eyes", "body", "tail"], "cheetah": ["head", "ears", "eyes", "body", "tail"], "brown_bear": ["head", "ears", "muzzle", "body", "legs"], "American_black_bear": ["head", "ears", "muzzle", "body", "legs"], "ice_bear": ["head", "ears", "muzzle", "body", "legs"], "sloth_bear": ["head", "ears", "muzzle", "body", "legs"], "mongoose": ["head", "ears", "muzzle", "body", "tail"], "meerkat": ["head", "ears", "eyes", "body", "tail"], "tiger_beetle": ["head", "thorax", "abdomen", "legs", "antennae"], "ladybug": ["head", "thorax", "wings", "legs", "antennae"], "ground_beetle": ["head", "thorax", "abdomen", "legs", "antennae"], "long-horned_beetle": ["head", "thorax", "abdomen", "long antennae", "legs"], "leaf_beetle": ["head", "thorax", "wings", "legs", "antennae"], "dung_beetle": ["head", "thorax", "abdomen", "legs"], "rhinoceros_beetle": ["head horn", "thorax", "abdomen", "legs"], "weevil": ["head snout", "thorax", "abdomen", "legs", "antennae"], "fly": ["head", "thorax", "abdomen", "wings", "legs"], "bee": ["head", "thorax", "abdomen", "wings", "legs"], "ant": ["head", "thorax", "abdomen", "antennae", "legs"], "grasshopper": ["head", "thorax", "abdomen", "hind legs", "wings"], "cricket": ["head", "thorax", "abdomen", "hind legs", "wings"], "walking_stick": ["head", "thorax", "abdomen", "legs"], "cockroach": ["head", "thorax", "abdomen", "legs", "antennae"], "mantis": ["head", "thorax", "abdomen", "raptorial forelegs", "wings"], "cicada": ["head", "thorax", "abdomen", "wings", "eyes"], "leafhopper": ["head", "thorax", "abdomen", "wings", "legs"], "lacewing": ["head", "thorax", "abdomen", "wings", "antennae"], "dragonfly": ["head", "thorax", "abdomen", "wings", "legs"], "damselfly": ["head", "thorax", "abdomen", "wings", "legs"], "admiral": ["head", "thorax", "abdomen", "wings", "antennae"], "ringlet": ["head", "thorax", "abdomen", "wings", "antennae"], "monarch": ["head", "thorax", "abdomen", "wings", "antennae"], "cabbage_butterfly": ["head", "thorax", "abdomen", "wings", "antennae"], "sulphur_butterfly": ["head", "thorax", "abdomen", "wings", "antennae"], "lycaenid": ["head", "thorax", "abdomen", "wings", "antennae"], "starfish": ["central disc", "arms", "tube feet"], "sea_urchin": ["test", "spines", "tube feet"], "sea_cucumber": ["body", "tentacles", "tube feet"], "wood_rabbit": ["head", "ears", "eyes", "body", "tail"], "hare": ["head", "ears", "eyes", "body", "tail"], "Angora_rabbit": ["head", "ears", "body", "legs", "tail"], "hamster": ["head", "ears", "eyes", "body", "tail"], "porcupine": ["head", "quills", "body", "legs"], "fox_squirrel": ["head", "ears", "body", "legs", "tail"], "marmot": ["head", "ears", "body", "legs", "tail"], "beaver": ["head", "teeth", "body", "tail", "legs"], "guinea_pig": ["head", "ears", "body", "legs", "tail"], "sorrel_horse": ["head", "mane", "body", "legs", "tail"], "zebra": ["head", "mane", "body", "legs", "tail"], "hog": ["head", "snout", "body", "legs", "tail"], "wild_boar": ["head", "tusks", "body", "legs", "tail"], "warthog": ["head", "tusks", "body", "legs", "tail"], "hippopotamus": ["head", "eyes and nostrils", "body", "legs", "tail"], "ox": ["head", "horns", "body", "legs", "tail"], "water_buffalo": ["head", "horns", "body", "legs", "tail"], "bison": ["head", "horns", "body", "legs", "tail"], "ram": ["head", "horns", "body", "legs", "tail"], "bighorn": ["head", "curved horns", "body", "legs", "tail"], "ibex": ["head", "horns", "body", "legs", "tail"], "hartebeest":

["head", "horns", "body", "legs", "tail"], "impala": ["head", "horns", "body", "legs", "tail"], "gazelle": ["head", "horns", "body", "legs", "tail"], "Arabian_camel": ["head", "hump", "neck", "legs", "tail"], "llama": ["head", "neck", "body", "legs", "tail"], "weasel": ["head", "body", "legs", "tail"], "mink": ["head", "body", "legs", "tail"], "polecat": ["head", "body", "legs", "tail"], "black-footed_ferret": ["head", "body", "legs", "tail"], "otter": ["head", "body", "legs", "tail"], "skunk": ["head", "body", "legs", "tail"], "badger": ["head", "body", "legs", "tail"], "armadillo": ["head", "armor plates", "legs", "tail"], "three-toed_sloth": ["head", "forelimbs", "hindlimbs", "claws"], "orangutan": ["head", "torso", "arms", "legs", "face"], "gorilla": ["head", "torso", "arms", "legs", "face"], "chimpanzee": ["head", "torso", "arms", "legs", "face"], "gibbon": ["head", "torso", "arms", "legs", "face"], "siamang": ["head", "torso", "arms", "legs", "face"], "guenon": ["head", "torso", "arms", "legs", "tail"], "patas_monkey": ["head", "torso", "arms", "legs", "tail"], "baboon": ["head", "torso", "arms", "legs", "tail"], "macaque": ["head", "torso", "arms", "legs", "tail"], "langur": ["head", "torso", "arms", "legs", "tail"], "colobus": ["head", "torso", "arms", "legs", "tail"], "proboscis_monkey": ["head", "nose", "torso", "arms", "legs"], "marmoset": ["head", "torso", "arms", "legs", "tail"], "capuchin": ["head", "torso", "arms", "legs", "tail"], "howler_monkey": ["head", "torso", "arms", "legs", "tail"], "titi": ["head", "torso", "arms", "legs", "tail"], "spider_monkey": ["head", "torso", "arms", "legs", "tail"], "squirrel_monkey": ["head", "torso", "arms", "legs", "tail"], "ring_tailed_monkey": ["head", "torso", "arms", "legs", "tail"], "indri_monkey": ["head", "torso", "arms", "legs", "tail"], "Indian_elephant": ["head", "trunk", "ears", "tusks", "legs"], "African_elephant": ["head", "trunk", "ears", "tusks", "legs"], "lesser_panda": ["head", "ears", "body", "legs", "tail"], "giant_panda": ["head", "ears", "body", "legs", "tail"], "barracouta": ["head", "body", "tail", "fins"], "eel": ["head", "body", "tail"], "coho": ["head", "body", "tail", "fins"], "rock_beauty": ["head", "body", "tail", "fins"], "anemone_fish": ["head", "body", "tail", "fins"], "sturgeon": ["head", "body", "tail", "fins"], "gar": ["head", "body", "tail", "fins"], "lionfish": ["head", "body", "tail", "fins", "spines"], "puffer": ["head", "body", "tail", "fins"], "abacus": ["frame", "rods", "beads"], "abaya": ["body", "sleeves", "hem", "front opening"], "academic_gown": ["yoke", "sleeves", "body", "hem"], "accordion": ["bellows", "keyboard", "bass buttons", "grille"], "acoustic_guitar": ["body", "neck", "fretboard", "headstock", "strings"], "aircraft_carrier": ["flight deck", "island", "hull", "elevator"], "airliner": ["nose", "fuselage", "wings", "tail", "engines"], "airship": ["envelope", "gondola", "fins", "propellers"], "altar": ["table", "top surface", "steps", "ornament"], "ambulance": ["cab", "patient compartment", "lights", "wheels"], "amphibian": ["head", "body", "limbs"], "analog_clock": ["face", "hands", "numbers", "bezel"], "apiary": ["hive boxes", "frames", "entrance"], "apron": ["body", "neck strap", "ties", "pocket"], "assault_rifle": ["stock", "barrel", "magazine", "sights"], "bakery": ["display case", "oven", "counter", "shelves"], "balance_beam": ["beam", "supports", "base"], "balloon": ["envelope", "knot", "string", "valve"], "banjo": ["body", "neck", "head", "strings"], "bannister": ["handrail", "balusters", "newel post"], "barbell": ["bar", "weight plates", "collars"], "barber_chair": ["seat", "backrest", "armrests", "footrest", "lever"], "barbershop": ["chair", "mirror", "sink", "counter"], "barn": ["roof", "walls", "doors", "loft"], "barometer": ["dial", "glass cover", "case"], "barrel": ["staves", "hoops", "heads"], "barrow": ["wheel", "tray", "handles", "legs"], "baseball": ["leather cover", "stitches", "core"], "basketball": ["surface", "seams"], "bassinet": ["frame", "mattress", "hood"], "bassoon": ["body", "crook", "bell", "keys"], "bathing_cap": ["cap body", "rim"],

"bathtub": ["bowl", "rim", "drain", "faucet"], "beach_wagon": ["bed", "wheels", "handle"], "beacon": ["light", "housing", "base"], "beaker": ["body", "lip", "base", "graduations"], "bearskin": ["fur cap", "chin strap", "rim"], "bell_cote": ["opening", "roof", "support"], "bib": ["body", "neck strap", "ties"], "bicycle-built-for-two": ["frame", "seats", "handlebars", "wheels", "chain"], "bikini": ["top", "bottom", "ties"], "birdhouse": ["entrance", "roof", "body", "perch"], "boathouse": ["slip", "roof", "walls", "door"], "bobsled": ["nose", "cockpit", "runners", "shell"], "bolo_tie": ["cord", "slide", "tips"], "bonnet": ["hood", "rim", "ties"], "bookcase": ["shelves", "back", "sides", "top"], "bookshop": ["shelves", "counter", "display", "entrance"], "bow": ["limb", "string", "grip", "arrow rest"], "bow_tie": ["knot", "wings", "band"], "brass": ["bell", "mouthpiece", "valves", "tubing"], "brassiere": ["cups", "band", "straps", "hook"], "breakwater": ["core", "armor blocks", "seaward face"], "breastplate": ["chest panel", "shoulder straps", "lower edge"], "buckle": ["frame", "prong", "bar"], "bulletproof_vest": ["front panel", "back panel", "shoulder straps"], "bullet_train": ["nose", "windows", "doors", "pantograph"], "butcher_shop": ["counter", "display case", "hooks", "scale"], "cab": ["roof", "doors", "windows", "wheel"], "canoe": ["bow", "stern", "hull", "thwarts"], "cardigan": ["body", "sleeves", "buttons", "collar"], "car_mirror": ["glass", "housing", "arm", "mount"], "carousel": ["platform", "poles", "animals", "canopy"], "carpenter_kit": ["hammer", "saw", "nails", "screwdriver"], "carton": ["walls", "top flap", "bottom flap"], "car_wheel": ["rim", "tire", "hub", "spokes"], "cash_machine": ["screen", "keypad", "cash slot", "card slot"], "cassette": ["case", "spool holes", "tape"], "cassette_player": ["cassette slot", "buttons", "display", "speaker"], "castle": ["tower", "wall", "gate", "keep"], "catamaran": ["hulls", "deck", "mast", "crossbeam"], "CD_player": ["tray", "buttons", "display"], "cello": ["body", "neck", "fingerboard", "strings", "bridge"], "chain": ["links", "end links"], "chainlink_fence": ["posts", "mesh", "top rail"], "chain_mail": ["rings", "coif", "hauberk"], "chain_saw": ["bar", "chain", "engine housing", "handle"], "chest": ["lid", "body", "handles", "lock"], "chiffonier": ["drawers", "top", "sides", "legs"], "chime": ["tubes", "frame", "striker"], "china_cabinet": ["glass doors", "shelves", "base", "top"], "Christmas_stocking": ["body", "cuff", "loop"], "church": ["tower", "nave", "roof", "entrance"], "cinema": ["screen", "seats", "aisle", "lobby"], "cliff_dwelling": ["rooms", "terraces", "walls"], "clog": ["upper", "sole", "heel"], "coffeepot": ["body", "spout", "handle", "lid"], "coil": ["turns", "core", "lead"], "confectionery": ["display", "counter", "shelves"], "container_ship": ["deck", "hull", "bridge", "containers"], "convertible": ["hood", "windshield", "seats", "wheels"], "corkscrew": ["spiral", "handle", "lever"], "cornet": ["bell", "mouthpiece", "valves", "tubing"], "cowboy_boot": ["upper", "heel", "toe", "pull tabs"], "cowboy_hat": ["brim", "crown", "band"], "cradle": ["frame", "rocker", "slats", "mattress"], "crib": ["slats", "mattress", "rail"], "Crock_Pot": ["pot", "lid", "handle", "base", "dial"], "croquet_ball": ["surface"], "crutch": ["handle", "shaft", "pad", "tip"], "cuirass": ["chest panel", "shoulder straps", "lower edge"], "dam": ["crest", "face", "spillway"], "desk": ["top", "drawers", "legs", "keyboard tray"], "dial_telephone": ["handset", "dial", "base", "cord"], "diaper": ["absorbent pad", "waistband", "fasteners"], "digital_clock": ["display", "buttons", "case"], "dining_table": ["top", "legs", "apron"], "dishwasher": ["door", "rack", "control panel", "spray arm"], "disk_brake": ["rotor", "caliper", "pad", "hub"], "dock": ["deck", "pilings", "cleats"], "dogsled": ["sled", "runners", "harness attachments"], "dome": ["shell", "drum", "opening"], "drum": ["shell", "head", "rim", "lugs"], "drumstick": ["shaft", "tip"], "Dutch_oven": ["body", "lid", "handles"], "electric_guitar": ["body", "neck", "pickups", "bridge",

"headstock"], "electric_locomotive": ["nose", "cab", "pantograph",
"bogies"], "entertainment_center": ["shelves", "cabinet", "openings"],
"face_powder": ["compact", "pad", "powder pan"], "feather_boa":
["feathers", "core"], "file": ["blade", "handle"], "fireboat": ["hull",
"deck", "water monitors", "superstructure"], "fire_engine": ["cab",
"pump panel", "ladders", "hose"], "fire_screen": ["frame", "mesh
panels", "stand"], "flagpole": ["pole", "finial", "halyard", "cleat"],
"flute": ["body", "headjoint", "keys", "footjoint"], "folding_chair":
["seat", "back", "frame", "hinge"], "football_helmet": ["shell", "face
mask", "chin strap", "padding"], "forklift": ["mast", "forks", "cab",
"counterweight"], "fountain": ["bowl", "nozzle", "basin"], "fountain_pen":
["nib", "barrel", "cap", "clip"], "four-poster": ["posts", "frame",
"canopy", "headboard"], "freight_car": ["body", "doors", "bogies"],
"French_horn": ["bell", "leadpipe", "rotary valves"], "fur_coat": ["body",
"collar", "sleeves", "lining"], "garbage_truck": ["cab", "hopper", "lift
mechanism", "wheels"], "gasmask": ["facepiece", "filters", "straps"],
"goblet": ["cup", "stem", "base"], "go-kart": ["chassis", "steering
wheel", "wheels", "seat"], "golf_ball": ["surface", "dimples"], "golfcart":
["body", "seats", "steering", "wheels"], "gondola": ["hull", "seat",
"oarlock"], "gong": ["disk", "suspension", "mallet"], "grand_piano":
["lid", "keyboard", "strings", "legs"], "greenhouse": ["glazing", "frame",
"benches", "vent"], "grocery_store": ["shelves", "checkout", "produce
display"], "guillotine": ["frame", "blade", "lunette"], "hair_slide":
["body", "clip"], "hair_spray": ["can", "nozzle", "cap"], "half_track":
["cab", "tracks", "wheels", "body"], "hand-held_computer": ["screen",
"buttons", "housing"], "handkerchief": ["cloth", "hem"], "hard_disc":
["platter", "spindle", "housing"], "harmonica": ["cover plates",
"comb", "reeds"], "harp": ["soundboard", "neck", "strings", "pedals"],
"harvester": ["header", "body", "wheels", "grain tank"], "hatchet":
["head", "handle"], "holster": ["pouch", "belt loop", "retention strap"],
"honeycomb": ["cells", "frame"], "hook": ["shank", "point", "eye"],
"hoopskirt": ["hoops", "fabric", "waistband"], "horizontal_bar": ["bar",
"supports", "base"], "horse_cart": ["body", "wheels", "shaft", "seat"],
"hourglass": ["upper bulb", "lower bulb", "narrow waist"], "iPod":
["screen", "wheel", "body", "case"], "jack-o-lantern": ["hollowed shell",
"carved face", "handle"], "jeep": ["grille", "windshield", "doors",
"wheels"], "jigsaw_puzzle": ["pieces", "image surface"], "jinrikisha":
["shaft", "seat", "wheels", "handles"], "joystick": ["stick", "base",
"buttons"], "knot": ["loop", "bight", "tail"], "lab_coat": ["body",
"sleeves", "pockets", "collar"], "lawn_mower": ["deck", "blade", "handle",
"wheels"], "lens_cap": ["cap body", "retention strap"], "library":
["shelves", "reading tables", "stacks"], "lifeboat": ["hull", "oars",
"seats", "lifelines"], "limousine": ["body", "windows", "doors", "wheels"],
"liner": ["hull", "deck", "superstructure"], "lotion": ["bottle",
"pump", "cap"], "loupe": ["lens", "frame", "handle"], "lumbermill":
["saw", "conveyor", "log deck"], "magnetic_compass": ["card", "housing",
"needle", "base"], "mailbag": ["body", "strap", "closure"], "mailbox":
["box", "door", "flag"], "maillot_jersey": ["body", "straps", "leg
openings"], "maillot_swimsuit": ["body", "straps", "leg openings"],
"manhole_cover": ["cover", "rim"], "maraca": ["body", "handle"],
"marimba": ["keys", "resonators", "frame", "mallets"], "mask": ["face
piece", "straps", "eye openings"], "maypole": ["pole", "ribbons",
"base"], "maze": ["paths", "walls", "entrance"], "medicine_chest":
["door", "shelves", "mirror"], "megalith": ["stone block", "capstone"],
"microphone": ["head", "body", "stand mount"], "military_uniform":
["jacket", "trousers", "insignia", "buttons"], "milk_can": ["body",
"lid", "handle"], "minibus": ["body", "doors", "windows", "wheels"],
"miniskirt": ["waistband", "skirt body", "hem"], "minivan": ["body",
"sliding door", "windows", "wheels"], "missile": ["warhead", "body",

"fins"], "mobile_home": ["body", "windows", "door", "skirting"], "Model_T":
["body", "wheels", "roof", "windshield"], "modem": ["ports", "indicator
lights", "case"], "monastery": ["cloister", "church", "cells"], "moped":
["frame", "engine", "seat", "wheels"], "mortar": ["barrel", "base plate",
"breech"], "mortarboard": ["board", "tassel", "band"], "mosque": ["dome",
"minaret", "prayer hall", "entrance"], "mosquito_net": ["mesh", "frame",
"suspension"], "motor_scooter": ["deck", "handlebar", "wheels", "seat"],
"mountain_bike": ["frame", "fork", "wheels", "handlebar"], "mousetrap":
["base", "spring", "bar", "trigger"], "moving_van": ["box", "cab", "door",
"lift"], "neck_brace": ["support band", "chin rest", "straps"], "nipple":
["teat", "base"], "notebook": ["cover", "pages", "binding"], "obelisk":
["shaft", "base", "capstone"], "oboe": ["body", "keys", "reed"], "ocarina":
["body", "mouthpiece", "finger holes"], "odometer": ["display", "gear
housing"], "oil_filter": ["canister", "filter media", "gasket"], "organ":
["pipes", "keyboard", "pedals", "case"], "oscilloscope": ["screen",
"controls", "case"], "overskirt": ["outer skirt", "waistband", "hem"],
"oxcart": ["bed", "wheels", "yoke"], "oxygen_mask": ["facepiece",
"straps", "hose"], "packet": ["wrapper", "seal"], "paddle": ["shaft",
"blade", "grip"], "paddlewheel": ["paddles", "axle", "housing"], "pajama":
["top", "bottom", "waistband"], "palace": ["facade", "towers", "gate",
"courtyard"], "panpipe": ["pipes", "binding"], "parachute": ["canopy",
"suspension lines", "harness"], "parallel_bars": ["bars", "supports",
"base"], "parking_meter": ["head", "dial", "post"], "passenger_car":
["body", "doors", "windows", "wheels"], "pay-phone": ["handset", "coin
slot", "dial", "housing"], "pedestal": ["platform", "shaft", "base"],
"Petri_dish": ["dish", "lid"], "photocopier": ["feeder", "glass platen",
"control panel", "output tray"], "pick": ["tip", "shaft", "handle"],
"pickelhaube": ["helmet shell", "spike", "chin strap"], "picket_fence":
["posts", "slats", "rails"], "pickup_truck": ["cab", "bed", "wheels",
"grille"], "pier": ["deck", "piles", "railings"], "piggy_bank": ["body",
"slot", "snout", "feet"], "ping-pong_ball": ["surface"], "pinwheel":
["blades", "hub", "stick"], "pirate": ["hat", "coat", "sash", "boot"],
"plane": ["nose", "wings", "fuselage", "tail"], "planetarium": ["dome",
"projector", "seating"], "plow": ["blade", "frame", "hitch", "wheels"],
"Polaroid_camera": ["body", "lens", "viewfinder", "film slot"], "pole":
["shaft", "base", "top"], "police_van": ["cab", "rear compartment",
"lights", "wheels"], "poncho": ["cape", "neck opening", "hem"],
"pool_table": ["playing surface", "rails", "pockets", "legs"], "pot":
["rim", "body", "handle", "base"], "potter_wheel": ["head", "foot pedal",
"wheel"], "prayer_rug": ["field", "border", "fringe"], "prison": ["walls",
"towers", "gates", "cells"], "missile_projectile": ["nose", "body",
"tail"], "projector": ["lens", "body", "mount", "controls"], "puck":
["disc"], "purse": ["body", "strap", "closure", "pocket"], "quill":
["shaft", "barbs", "tip"], "quilt": ["patches", "binding", "stitch lines"],
"racer": ["cockpit", "body", "wheels", "wing"], "radiator": ["fins",
"inlet", "outlet", "core"], "radio": ["speaker", "tuning dial", "antenna",
"housing"], "radio_telescope": ["dish", "feed", "support structure"],
"rain_barrel": ["body", "inlet", "spigot", "lid"], "recreational_vehicle":
["cab", "living area", "windows", "wheels"], "reel": ["spool", "handle",
"frame"], "reflex_camera": ["body", "lens", "viewfinder", "shutter"],
"refrigerator": ["door", "shelves", "handle", "compressor"], "restaurant":
["entrance", "dining area", "counter", "kitchen view"], "revolver":
["barrel", "cylinder", "grip", "trigger"], "rifle": ["barrel", "stock",
"sight", "trigger"], "rocking_chair": ["seat", "back", "rockers",
"armrests"], "rotisserie": ["spit", "motor", "frame", "drip tray"],
"rubber_eraser": ["body", "edge"], "rugby_ball": ["panel", "seams"],
"safe": ["door", "dial", "body", "hinge"], "sarong": ["wrap", "waist
tie", "hem"], "sax": ["body", "neck", "mouthpiece", "keys"], "school_bus":
["front", "windows", "doors", "wheels"], "schooner": ["hull", "masts",

"sails", "deck"], "screen": ["frame", "mesh", "mount"], "screwdriver":
["shaft", "tip", "handle"], "sewing_machine": ["needle area", "bed", "hand
wheel", "foot pedal"], "shield": ["face", "rim", "handle"], "shoe_shop":
["shelves", "display", "counter"], "shoji": ["frame", "paper panels",
"sliding track"], "shopping_cart": ["basket", "handle", "wheels"],
"shower_curtain": ["curtain panel", "grommets", "rod"], "ski": ["tip",
"camber", "binding", "tail"], "ski_mask": ["face opening", "eye openings",
"edge"], "slide_rule": ["body", "slider", "scales"], "sliding_door":
["panel", "track", "handle"], "slot": ["opening", "face", "controls"],
"snowmobile": ["front ski", "track", "seat", "handlebar"], "snowplow":
["blade", "frame", "hydraulics", "mount"], "soccer_ball": ["panels",
"seams"], "solar_dish": ["dish", "receiver", "mount"], "sombrero":
["crown", "brim", "band"], "space_bar": ["keycap"], "space_shuttle":
["nose", "fuselage", "wings", "tail"], "speedboat": ["hull", "deck",
"windshield", "engine"], "spider_web": ["radial threads", "spiral threads",
"anchor points"], "spindle": ["shaft", "whorl", "hook"], "sports_car":
["body", "windshield", "wheels", "spoiler"], "spotlight": ["lamp",
"reflector", "housing", "mount"], "stage": ["platform", "backdrop",
"human", "lighting"], "steam_locomotive": ["boiler", "cab", "tender",
"wheels"], "steel_arch_bridge": ["arch", "deck", "support piers"],
"steel_drum": ["body", "lip", "notes area"], "stethoscope": ["chest
piece", "tubing", "earpieces"], "stole": ["strip", "ends"], "stone_wall":
["stones", "mortar", "cap"], "stopwatch": ["face", "buttons", "case"],
"stove": ["burners", "control knobs", "oven door"], "streetcar":
["body", "windows", "doors", "wheels"], "stretcher": ["frame", "canvas",
"handles", "wheels"], "studio_couch": ["seat", "back", "arms", "legs"],
"stupa": ["dome", "harmika", "base"], "submarine": ["conning tower",
"hull", "propeller", "rudder"], "sundial": ["dial plate", "gnomon",
"base"], "sunglass": ["lens", "frame", "temples"], "sunscreen": ["tube",
"cap", "dispensing nozzle"], "suspension_bridge": ["towers", "cables",
"deck"], "swing": ["seat", "chains", "support"], "switch": ["toggle",
"plate", "mount"], "tank": ["turret", "body", "tracks", "gun barrel"],
"tape_player": ["reels", "controls", "play head"], "tennis_ball":
["surface"], "thatch": ["bundles", "ridge", "eaves"], "theater_curtain":
["curtain panel", "hem", "pelmet"], "thimble": ["cap", "rim"], "thresher":
["drum", "feeder", "frame"], "throne": ["seat", "back", "armrests",
"base"], "tile_roof": ["tiles", "ridge", "eaves"], "tobacco_shop":
["shelves", "display", "counter"], "toilet_seat": ["seat", "lid",
"hinges"], "torch": ["head", "body", "switch", "lens"], "tow_truck":
["cab", "boom", "wheels", "winch"], "toyshop": ["shelves", "display",
"counter"], "tractor": ["body", "wheels", "cab", "three point hitch"],
"trailer_truck": ["cab", "trailer", "wheels", "doors"], "trench_coat":
["collar", "buttons", "belt", "sleeves"], "tricycle": ["frame", "front
wheel", "rear wheels", "seat"], "trimaran": ["center hull", "amas",
"deck", "mast"], "tripod": ["legs", "head", "mount"], "triumphal_arch":
["archway", "column", "entablature"], "trombone": ["slide", "bell",
"mouthpiece"], "tub": ["rim", "bowl", "drain"], "turnstile": ["arms",
"shaft", "housing"], "typewriter_keyboard": ["keycaps", "carriage", "space
bar"], "unicycle": ["wheel", "seat", "pedal"], "upright_piano": ["body",
"keyboard", "strings", "pedal"], "vault": ["door", "chamber", "locking
mechanism"], "velvet": ["pile", "edge"], "vending_machine": ["front
panel", "selection buttons", "dispense slot"], "viaduct": ["arches",
"deck", "piers"], "violin": ["body", "neck", "fingerboard", "strings",
"bridge"], "volleyball": ["panels", "seams"], "waffle_iron": ["plates",
"hinge", "handle"], "wall_clock": ["face", "hands", "case"], "wardrobe":
["doors", "rails", "drawers", "top"], "warplane": ["nose", "wings",
"fuselage", "engines"], "washbasin": ["bowl", "tap", "overflow"], "washer":
["drum", "door", "control panel"], "water_jug": ["neck", "body", "handle",
"base"], "water_tower": ["tank", "support", "ladder"], "wig": ["cap",

"hair"], "window_screen": ["frame", "mesh", "spline"], "window_shade": ["shade panel", "roller", "cord"], "Windsor_tie": ["knot", "blade", "tail"], "wing": ["airfoil", "flap", "slat"], "wool": ["fibre", "bundle"], "worm_fence": ["posts", "rails", "weave"], "wreck": ["bow", "stern", "hull"], "yawl": ["hull", "mast", "sails", "rudder"], "yurt": ["wall", "roof ring", "cover", "door"], "web_site": ["header", "navigation", "content area", "footer"], "crossword_puzzle": ["grid", "clues", "filled squares"], "street_sign": ["post", "sign panel", "mount"], "traffic_light": ["lanes", "lights", "housing"], "menu": ["cover", "pages", "items"], "guacamole": ["bowl", "dip"], "consomme": ["bowl", "liquid"], "hot_pot": ["pot", "broth", "ingredients"], "trifle": ["layers", "glass", "topping"], "ice_cream": ["scoop", "cone", "topping"], "bagel": ["crust", "hole", "surface"], "pretzel": ["twist", "surface"], "cheeseburger": ["bun", "patty", "cheese", "toppings"], "hotdog": ["bun", "sausage", "toppings"], "mashed_potato": ["pile", "scoop"], "head_cabbage": ["outer leaves", "core", "head"], "broccoli": ["head", "stalk", "florets"], "cauliflower": ["curd", "stem", "florets"], "zucchini": ["stem end", "body", "blossom end"], "spaghetti_squash": ["skin", "flesh", "seeds"], "acorn_squash": ["skin", "flesh", "stem"], "butternut_squash": ["skin", "flesh", "stem"], "cucumber": ["stem end", "body", "blossom end"], "artichoke": ["head", "bracts", "stem"], "bell_pepper": ["stem", "body", "core"], "cardoon": ["stem", "leaves"], "mushroom": ["cap", "gills", "stem"], "Granny_Smith": ["skin", "flesh", "stem"], "strawberry": ["body", "calyx", "seeds"], "fig": ["skin", "flesh", "seeds"], "pineapple": ["crown", "skin", "core"], "jackfruit": ["skin", "flesh pods", "seeds"], "custard_apple": ["skin", "segments", "seeds"], "pomegranate": ["crown", "arils", "rind"], "hay": ["stalks", "bales"], "chocolate_sauce": ["bottle", "pour stream"], "dough": ["surface", "edges"], "meat_loaf": ["slice", "crust"], "pizza": ["crust", "cheese", "toppings", "slice"], "potpie": ["crust", "filling"], "burrito": ["tortilla", "filling", "ends"], "red_wine": ["glass", "liquid"], "espresso": ["cup", "crema"], "eggnog": ["cup", "foam"], "alp": ["peak", "slope", "snow cap"], "bubble": ["film", "reflection"], "cliff": ["face", "edge", "base"], "coral_reef": ["reef crest", "patches", "channels"], "geyser": ["vent", "plume", "pool"], "lakeside": ["shore", "water", "vegetation"], "promontory": ["headland", "cliff", "shoreline"], "sandbar": ["crest", "sand", "shallow water"], "seashore": ["beach", "waves", "shoreline"], "valley": ["floor", "sides", "stream"], "volcano": ["cone", "crater", "lava flow"], "ballplayer": ["helmet", "jersey", "glove"], "groom": ["suit", "tie", "bouquet"], "scuba_diver": ["mask", "tank", "fins", "wetsuit"], "rapeseed": ["flowers", "stalks", "pods"], "daisy": ["disk", "petals", "stem"], "yellow_lady_slipper": ["pouch", "sepals", "stem"], "corn": ["ear", "kernels", "husks"], "acorn": ["cap", "nut"], "hip": ["fruit", "sepals"], "buckeye": ["shell", "nut"], "coral_fungus": ["branches", "tips"], "agaric": ["cap", "gills", "stem"], "gyromitra": ["cap", "stem"], "stinkhorn": ["stalk", "cap", "gleba"], "earthstar": ["central disc", "rays"], "hen-of-the-woods": ["clusters", "caps"], "bolete": ["cap", "pore surface", "stem"], "corn_ear": ["kernels", "cob", "husk", "lobe", "corn"] }

ImageNet/ObjectNet-54 Class names:

['snorkel', 'loudspeaker', 'syringe', 'binoculars', 'cannon', 'computer_keyboard', 'crane_machine', 'desktop_computer', 'gas_pump', 'remote_control', 'screw', 'muzzle', 'seat_belt', 'patio', 'home_theater', 'totem_pole', 'grille', 'mountain_tent', 'scoreboard', 'plate_rack', 'ashcan', 'cocktail_shaker', 'kimono', 'whiskey_jug', 'knee_pad', 'book_jacket', 'crash_helmet', 'vestment', 'cloak', 'scabbard', 'beer_glass', 'swab', 'drilling_platform', 'sweatshirt', 'pencil_box', 'punching_bag', 'pencil_sharpener', 'broom', 'cup', 'shower_cap', 'envelope', 'trolleybus', 'perfume', 'crate', 'ballpoint', 'comic_book',

'shopping_basket', 'wooden_spoon', 'ice_lolly', 'carbonara', 'paintbrush',
'iron', 'toaster', 'caldron']

ImageNet/ObjectNet-104 Class names:

['backpack', 'banana', 'Band_Aid', 'shopping_basket', 'bath_towel',
'beer_bottle', 'park_bench', 'binder', 'bottlecap', 'French_loaf', 'broom',
'bucket', 'cleaver', 'can_opener', 'candle', 'cellular_telephone', 'hamper',
'espresso_maker', 'combination_lock', 'mouse', 'table_lamp', 'dishrag',
'doormat', 'Loafer', 'power_drill', 'cup', 'plate_rack', 'envelope',
'electric_fan', 'frying_pan', 'gown', 'hand_blower', 'hammer', 'iron',
'jean', 'computer_keyboard', 'ladle', 'lampshade', 'laptop', 'lemon',
'letter_opener', 'lighter', 'lipstick', 'matchstick', 'measuring_cup',
'microwave', 'mixing_bowl', 'monitor', 'coffee_mug', 'nail', 'necklace',
'orange', 'padlock', 'paintbrush', 'paper_towel', 'pill_bottle', 'pillow',
'pitcher', 'plastic_bag', 'plate', 'plunger', 'pop_bottle', 'space_heater',
'printer', 'remote_control', 'rule', 'running_shoe', 'safety_pin',
'saltshaker', 'sandal', 'screw', 'shovel', 'sleeping_bag', 'soap_dispenser',
'sock', 'soup_bowl', 'spatula', 'loudspeaker', 'strainer', 'teddy', 'suit',
'sunglasses', 'sweatshirt', 'swimming_trunks', 'jersey', 'television',
'teapot', 'racket', 'toaster', 'toilet_tissue', 'ashcan', 'tray',
'umbrella', 'vacuum', 'vase', 'wallet', 'digital_watch', 'water_bottle',
'dumbbell', 'scale', 'whistle', 'wine_bottle', 'mitten', 'wok']

CUB

ID Class names: ['Sooty_Albatross', 'Crested_Auklet', 'Least_Auklet',
'Brewer_Blackbird', 'Rusty_Blackbird', 'Indigo_Bunting', 'Spotted_Catbird',
'Brandt_Cormorant', 'Bronzed_Cowbird', 'American_Crow', 'Mangrove_Cuckoo',
'Purple_Finch', 'Acadian_Flycatcher', 'Least_Flycatcher',
'Vermilion_Flycatcher', 'American_Goldfinch', 'Eared_Grebe', 'Horned_Grebe',
'Evening_Grosbeak', 'Pine_Grosbeak', 'California_Gull', 'Heermann_Gull',
'Herring_Gull', 'Ivory_Gull', 'Western_Gull', 'Anna_Hummingbird',
'Pomarine_Jaeger', 'Florida_Jay', 'Tropical_Kingbird', 'Belted_Kingfisher',
'Pied_Kingfisher', 'Hooded_Merganser', 'Baltimore_Oriole',
'Hooded_Oriole', 'Brown_Pelican', 'Common_Raven', 'Loggerhead_Shrike',
'Baird_Sparrow', 'Brewer_Sparrow', 'Fox_Sparrow', 'Grasshopper_Sparrow',
'Harris_Sparrow', 'Henslow_Sparrow', 'Lincoln_Sparrow', 'Savannah_Sparrow',
'Song_Sparrow', 'Vesper_Sparrow', 'Barn_Swallow', 'Cliff_Swallow',
'Scarlet_Tanager', 'Artic_Tern', 'Caspian_Tern', 'Common_Tern',
'Sage_Thrasher', 'Philadelphia_Vireo', 'Warbling_Vireo', 'Canada_Warbler',
'Cerulean_Warbler', 'Kentucky_Warbler', 'Magnolia_Warbler',
'Mourning_Warbler', 'Myrtle_Warbler', 'Nashville_Warbler',
'Prairie_Warbler', 'Prothonotary_Warbler', 'Swainson_Warbler',
'Tennessee_Warbler', 'Wilson_Warbler', 'Louisiana_Waterthrush',
'Bohemian_Waxwing', 'Pileated_Woodpecker', 'Downy_Woodpecker',
'Bewick_Wren', 'Carolina_Wren']

Coarse-grained OOD Class names:

['White_breasted_Nuthatch', 'Horned_Puffin', 'Sayornis',
'Northern_Fulmar', 'Brown_Creeper', 'Green_tailed_Towhee', 'Mallard',
'Northern_Flicker', 'Ovenbird', 'Chuck_will_Widow', 'Nighthawk',
'Cardinal', 'Bobolink', 'Geococcyx', 'Gadwall', 'American_Pipit',
'Whip_poor_Will', 'Green_Violetear', 'Cape_Glossy_Starling',
'Yellow_breasted_Chat', 'Western_Wood_Pewee', 'Eastern_Towhee',
'Frigatebird', 'Boat_tailed_Grackle', 'Western_Meadowlark',
'Pigeon_Guillemot', 'Clark_Nutcracker', 'Dark_eyed_Junco',
'American_Redstart', 'Common_Yellowthroat', 'Pacific_Loon', 'Mockingbird',
'Horned_Lark', 'Groove_billed_Ani', 'Red_legged_Kittiwake']

Fine-grained OOD Class names:

['Northern_Waterthrush', 'Marsh_Wren', 'White_eyed_Vireo',
'Blue_Jay', 'European_Goldfinch', 'Cactus_Wren', 'Cedar_Waxwing',
'Nelson_Sharp_tailed_Sparrow', 'Tree_Sparrow', 'Rufous_Hummingbird',
'Shiny_Cowbird', 'White_Pelican', 'Black_billed_Cuckoo',
'Scissor_tailed_Flycatcher', 'Yellow_headed_Blackbird',
'Blue_winged_Warbler', 'Red_winged_Blackbird', 'Gray_Kingbird',
'Western_Grebe', 'Le_Conte_Sparrow', 'White_breasted_Kingfisher',
'Fish_Crow', 'Chipping_Sparrow', 'House_Wren', 'Tree_Swallow',
'Long_tailed_Jaeger', 'Red_eyed_Vireo', 'Red_faced_Cormorant',
'Ringed_Kingfisher', 'Winter_Wren', 'Black_and_white_Warbler',
'Rhinoceros_Auklet', 'Field_Sparrow', 'Black_throated_Blue_Warbler',
'Yellow_throated_Vireo', 'Slaty_backed_Gull', 'Red_bellied_Woodpecker',
'Bank_Swallow', 'Bay_breasted_Warbler', 'Rose_breasted_Grosbeak',
'Green_Jay', 'Pelagic_Cormorant', 'Glaucous_winged_Gull',
'White_necked_Raven', 'Summer_Tanager', 'White_throated_Sparrow',
'Green_Kingfisher', 'Black_Tern', 'Hooded_Warbler', 'Yellow_Warbler',
'Ring_billed_Gull', 'Lazuli_Bunting', 'Laysan_Albatross',
'Olive_sided_Flycatcher', 'Seaside_Sparrow', 'Parakeet_Auklet',
'Gray_crowned_Rosy_Finch', 'Gray_Catbird', 'Orange_crowned_Warbler',
'Pied_billed_Grebe', 'Red_breasted_Merganser', 'Yellow_bellied_Flycatcher',
'Painted_Bunting', 'Black_throated_Sparrow', 'Yellow_billed_Cuckoo',
'Orchard_Oriole', 'Forsters_Tern', 'Pine_Warbler', 'White_crowned_Sparrow',
'Black_footed_Albatross', 'Black_capped_Vireo', 'Great_Grey_Shrike',
'Blue_Grosbeak', 'Rock_Wren', 'Palm_Warbler', 'House_Sparrow',
'Least_Tern', 'Worm_eating_Warbler', 'Clay_colored_Sparrow',
'Chestnut_sided_Warbler', 'Golden_winged_Warbler', 'Scott_Oriole',
'Red_headed_Woodpecker', 'Cape_May_Warbler', 'Blue_headed_Vireo',
'American_Three_toed_Woodpecker', 'Red_cockaded_Woodpecker',
'Ruby_throated_Hummingbird', 'Great_Crested_Flycatcher', 'Elegant_Tern',
'Brown_Thrasher']

2-Component names:

['head', 'body']

3-Component names:

['head', 'body', 'legs']

4-Component names:

['head', 'body', 'wing', 'legs']

5-Component names:

['head', 'body', 'wing', 'legs', 'tail']

12-Component names:

['back', 'beak', 'belly', 'breast', 'crown', 'forehead', 'eye', 'leg',
'wing', 'nape', 'tail', 'throat']

F.3 Non-textual component construction

We follow the concept extraction and mask assignment procedures of ACE [53] with its default settings. Specifically, we apply multi-scale segmentation, followed by K-means clustering and outlier removal to obtain visual concept prototypes. Component masks are obtained by assigning image superpixels to their nearest discovered concepts in feature space. These masks replace GradCAM-based localization while retaining non-component suppression in CoOD. We further discard textual component prototypes and directly use CLIP visual features of the suppressed ID regions as component prototypes, against which test component representations are compared.

G FPR reduction at small component numbers

Constrained by the finite capacity of visual encoders to represent extremely fine-grained or low-resolution components, we consider the applicability of the normal approximation when the number of components is small.

We retain the notation from the main paper. For each component index p define the existence score as $s_{yp} = \langle \boldsymbol{z}_{yp}, \boldsymbol{t}_{yp} \rangle$, and model the indicator $\mathbb{1}\{s_{yp} > \gamma\}$ as a Bernoulli variable with probability of being an ID component $\psi = P(s_{yp} > \gamma | \boldsymbol{x})$. The aggregated score from $|\mathcal{P}_y|$ components $\mathcal{S}_{|\mathcal{P}_y|} = \sum_{p=1}^{|\mathcal{P}_y|} \mathbb{1}\{s_{yp} > \gamma\}$ follows a Binomial distribution $\mathcal{B}(|\mathcal{P}_y|, \psi)$. Fix a target ID true positive rate λ, for each component number $|\mathcal{P}_y|$, let $T_{|\mathcal{P}_y|} \in \mathbb{Z}_{\geq 0}$ denote the integer threshold chosen by the minimal integer satisfying the TPR constraint, *i.e.* $T_{|\mathcal{P}_y|} = \min\left\{t \in \mathbb{Z} : \mathbb{P}\big(\mathcal{S}_{|\mathcal{P}_y|} > t \mid \text{ID}\big) \leq \lambda\right\}$. The corresponding false positive rate FPR_λ is given by:

$$\text{FPR}_\lambda(|\mathcal{P}_y|) = P(\mathcal{S}_{|\mathcal{P}_y|} > T_{|\mathcal{P}_y|} | \text{OOD}) = \sum_{p > T_{|\mathcal{P}_y|}}^{|\mathcal{P}_y|} \binom{|\mathcal{P}_y|}{p} (\psi^{\text{out}})^p (1 - \psi^{\text{out}})^{|\mathcal{P}_y| - p}. \tag{16}$$

For an arbitrary integer T, the binomial recursion gives:

$$P(\mathcal{S}_{|\mathcal{P}_y|+1} > T \mid \text{OOD}) = (1 - \psi^{\text{out}})\, P(\mathcal{S}_{|\mathcal{P}_y|} > T \mid \text{OOD}) + \psi^{\text{out}}\, P(\mathcal{S}_{|\mathcal{P}_y|} > T - 1 \mid \text{OOD}). \tag{17}$$

We consider the change in FPR when adding one component:

$$\Delta = \text{FPR}_\lambda(|\mathcal{P}_y| + 1) - \text{FPR}_\lambda(|\mathcal{P}_y|) = P(\mathcal{S}_{|\mathcal{P}_y|+1} > T_{|\mathcal{P}_y|+1} | \text{OOD}) - P(\mathcal{S}_{|\mathcal{P}_y|} > T_{|\mathcal{P}_y|} | \text{OOD}) \tag{18}$$

Applying Equation (17) with $T = T_{|\mathcal{P}_y|+1}$ and expanding the binomial sums yields:

$$\begin{aligned}
\Delta &= (1 - \psi^{\text{out}}) P(\mathcal{S}_{|\mathcal{P}_y|} > T_{|\mathcal{P}_y|+1} | \text{OOD}) + \psi^{\text{out}} P(\mathcal{S}_{|\mathcal{P}_y|} > T_{|\mathcal{P}_y|+1} - 1 | \text{OOD}) \\
&\quad - P(\mathcal{S}_{|\mathcal{P}_y|} > T_{|\mathcal{P}_y|} | \text{OOD}) \\
&= (1 - \psi^{\text{out}}) \sum_{p=T_{|\mathcal{P}_y|+1}+1}^{|\mathcal{P}_y|} \binom{|\mathcal{P}_y|}{p} (\psi^{\text{out}})^p (1 - \psi^{\text{out}})^{|\mathcal{P}_y| - p} \\
&\quad + \psi^{\text{out}} \sum_{p=T_{|\mathcal{P}_y|+1}}^{|\mathcal{P}_y|} \binom{|\mathcal{P}_y|}{p} (\psi^{\text{out}})^p (1 - \psi^{\text{out}})^{|\mathcal{P}_y| - p} \\
&\quad - \sum_{p=T_{|\mathcal{P}_y|}+1}^{|\mathcal{P}_y|} \binom{|\mathcal{P}_y|}{p} (\psi^{\text{out}})^p (1 - \psi^{\text{out}})^{|\mathcal{P}_y| - p} \\
&= \sum_{p=T_{|\mathcal{P}_y|+1}+1}^{|\mathcal{P}_y|} \binom{|\mathcal{P}_y|}{p} (\psi^{\text{out}})^p (1 - \psi^{\text{out}})^{|\mathcal{P}_y| - p} + \psi^{\text{out}} \sum_{p=T_{|\mathcal{P}_y|+1}}^{T_{|\mathcal{P}_y|+1}} \binom{|\mathcal{P}_y|}{p} (\psi^{\text{out}})^p (1 - \psi^{\text{out}})^{|\mathcal{P}_y| - p} \\
&\quad - \sum_{p=T_{|\mathcal{P}_y|+1}+1}^{|\mathcal{P}_y|} \binom{|\mathcal{P}_y|}{p} (\psi^{\text{out}})^p (1 - \psi^{\text{out}})^{|\mathcal{P}_y| - p} - \sum_{p=T_{|\mathcal{P}_y|}+1}^{T_{|\mathcal{P}_y|+1}} \binom{|\mathcal{P}_y|}{p} (\psi^{\text{out}})^p (1 - \psi^{\text{out}})^{|\mathcal{P}_y| - p},
\end{aligned} \tag{19}$$

where the first and third terms cancel out, the second term has $p = T_{|\mathcal{P}_y|+1}$, and the fourth term exists only if $T_{|\mathcal{P}_y|+1} \geq T_{|\mathcal{P}_y|} + 1$. Evaluate Δ in the two possible cases, *i.e.*,

$$\Delta = \begin{cases}
\psi^{\text{out}} \binom{|\mathcal{P}_y|}{T_{|\mathcal{P}_y|+1}} (\psi^{\text{out}})^{T_{|\mathcal{P}_y|+1}} (1 - \psi^{\text{out}})^{|\mathcal{P}_y| - T_{|\mathcal{P}_y|+1}} \geq 0, & T_{|\mathcal{P}_y|+1} = T_{|\mathcal{P}_y|}, \\
(\psi^{\text{out}} - 1) \binom{|\mathcal{P}_y|}{T_{|\mathcal{P}_y|+1}} (\psi^{\text{out}})^{T_{|\mathcal{P}_y|+1}} (1 - \psi^{\text{out}})^{|\mathcal{P}_y| - T_{|\mathcal{P}_y|+1}} & \\
\quad - \sum_{p=T_{|\mathcal{P}_y|}+1}^{T_{|\mathcal{P}_y|+1}-1} \binom{|\mathcal{P}_y|}{p} (\psi^{\text{out}})^p (1 - \psi^{\text{out}})^{|\mathcal{P}_y| - p} \leq 0, & T_{|\mathcal{P}_y|+1} \geq T_{|\mathcal{P}_y|} + 1.
\end{cases} \tag{20}$$

When $\psi^{\text{out}} < \psi^{\text{in}} \leq 1$, $\Delta < 0$ in the second case. That is, adding a component tends to increase the FPR_λ if the threshold $T_{|\mathcal{P}_y|+1}$ does not move, and the FPR_λ decreases when the threshold increases.

Then we link this formulation to ψ^{in}. The threshold update occurs when the ID-side probability with $|\mathcal{P}_y| + 1$ components exceeds the previous threshold λ, *i.e.*

$$P(\mathcal{S}_{|\mathcal{P}_y|+1} > T_{|\mathcal{P}_y|}|\text{ID}) = (1-\psi^{\text{in}})P(\mathcal{S}_{|\mathcal{P}_y|} > T_{|\mathcal{P}_y|}|\text{ID}) + \psi^{\text{in}}P(\mathcal{S}_{|\mathcal{P}_y|} > T_{|\mathcal{P}_y|} - 1|\text{ID}) > \lambda. \tag{21}$$

Note that Equation (21) increases monotonically *w.r.t.* ψ^{in}, because $P(\mathcal{S}_{|\mathcal{P}_y|} > T_{|\mathcal{P}_y|}|\text{ID}) < P(\mathcal{S}_{|\mathcal{P}_y|} > T_{|\mathcal{P}_y|} - 1|\text{ID})$ in practice. Therefore, a larger ψ^{in} increases the probability of the threshold to increase, which reduces FPR when the update occurs. Taking expectation over multi-step threshold update gives that, in aggregate, an increase in ψ^{in} together with a decrease in ψ^{out} is more likely to make FPR_λ decrease after introducing a new component.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: See Section 1.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: See Section 7.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

Justification: See theoretical results in Section 3.2 and Appendix G.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: See Section 4

   Guidelines:

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

5. **Open access to data and code**

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

Answer: [Yes]

Justification: We will release the code.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: See Section 4.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: See Appendix E.5 and Section 4.3.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: See Section 4.3.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We conformed the code.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: See Section 6.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: No such risks.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: See Reference Section.

    Guidelines:

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

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [N/A].

    Justification: No new asset.

    Guidelines:

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

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

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: No crowdsourcing and research with human subject.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

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

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No research with human subject.

    Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [Yes]

Justification: All usage procedures, prompts, and outputs have been provided.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.