SAGE: Semantic-Anchor-Guided Embedding for Generalized AI-Generated Image Detection

Seongho Kim **Jaehyun Choi** **Dahye Kim** **Sungwon Yi** **Jang-Ho Choi***
AI Safety Institute, ETRI
South Korea
{sbde500, pre6ent, dhaisi, sungyi, janghochoi}@etri.re.kr

Abstract

AI-generated image detection has become an important problem in media forensics as modern generative models produce increasingly realistic images. Recent CLIP-based detectors show strong generalization ability, but CLIP image features entangle semantic content with forensic cues. Since semantic content is not an intrinsic generation trace, relying on it can introduce semantic shortcuts and unstable performance under unseen generator shifts. In this paper, we propose **SAGE**[2] (Semantic-Anchor-Guided Embedding), a framework that learns authentication-oriented representations with reduced dependence on caption-level semantics. SAGE constructs paired triplets of COCO images, SDXL-reconstructed images, and COCO captions so that real and fake images share the same semantics. It reduces dependence on caption-level semantics by learning from normalized image–text residuals with a projection module and a decorrelation objective. To enable caption-free inference, SAGE learns a semantic anchor that approximates the role of paired text embeddings at test time. In addition, SAGE adopts a SoftTriple-style sub-center classifier to model intra-class variation in real and fake images, yielding a more stable authentication geometry across diverse generators. Experiments on CommunityForensics demonstrate that SAGE achieves competitive detection performance while substantially reducing generator-wise performance variation. Further analyses show that SAGE effectively reduces semantic dependence and forms a clearer real/fake feature geometry than strong baselines.

1 Introduction

The rapid advancement of image generative models has fundamentally changed the way digital visual content is produced. Generative adversarial networks (GANs) [6, 13], autoregressive models [35, 28], diffusion models [11, 5], and recent text-to-image foundation models [29, 24] can now generate not only high-quality natural images but also synthetic images that are increasingly difficult to distinguish from real ones. While these advances expand the potential of generative models as creative tools, they also raise serious societal concerns, including misinformation, copyright infringement, identity forgery, and manipulation of digital evidence. Therefore, AI-generated image detection, which aims to determine whether an input image is real or synthesized by a generative model, has become an important problem in modern media forensics.

Many studies on AI-generated image detection have sought to prevent the misuse of generative models by identifying traces left by generators [20, 41]. However, existing methods have still been reported to lack sufficient detection capability for unseen generators [10, 39]. To address this limitation, recent studies have explored CLIP [27], which provides rich semantic representations through large-scale

*Corresponding author.
[2]Code is available at KoreaAISI-SAGE.

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

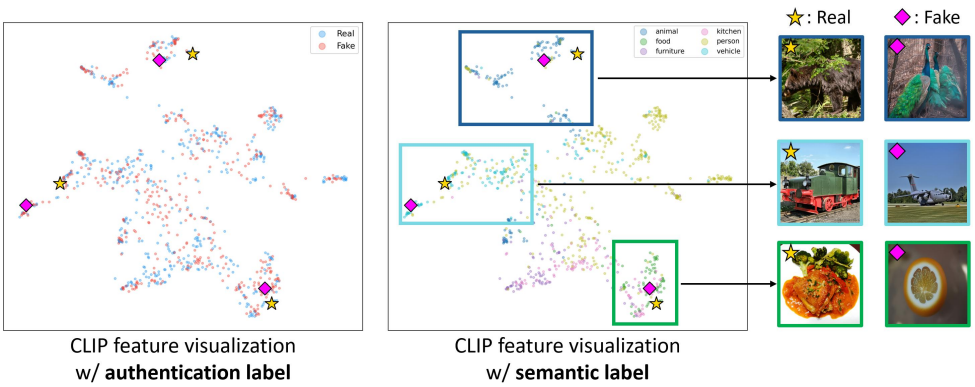

Figure 1: Visualization of CLIP image embeddings for real and fake images. Left: UMAP projection colored by real/fake labels. Right: UMAP projection of the same image set colored by content categories.

contrastive learning [39, 14]. Nevertheless, it remains unclear whether CLIP semantics are always beneficial for real/fake authentication, especially under shifts caused by commercial or previously unseen generators. This motivates us to investigate the following two questions: **1)** although CLIP provides useful representations, are its features truly useful for real/fake authentication? **2)** since not only generators but also real images come from diverse sources, is binary classification alone sufficient to distinguish them reliably? To explore the first question, we analyze CLIP image embeddings. Figure 1 shows the result. The left plot visualizes CLIP image features for real and fake images using UMAP, while the right plot maps the same image set according to content categories. Even when two images differ in terms of being real or fake, if they share similar semantics, the CLIP image encoder tends to place them in nearby regions. This observation does not imply that CLIP features contain no forensic cues; rather, it indicates that their neighborhood geometry is strongly influenced by semantic content. Such semantic dominance can interfere with learning authentication-oriented decision boundaries, since semantic content is not an intrinsic generation trace and its association with the real/fake label may depend on the training data or prompt distribution.

To address the second question, we focus on metric learning studies that represent a single class with multiple sub-centers. A conventional softmax classifier represents each class with a single weight vector or prototype, making it difficult to sufficiently model diverse local modes within a class. To alleviate this limitation, SoftTriple introduces multiple learnable centers for each class and performs metric learning without triplet sampling by softly assigning each sample to the most suitable center within its class [26]. Similarly, Sub-center ArcFace assigns K sub-centers to each class, allowing samples to be pulled toward one of multiple positive sub-centers rather than a single positive center, thereby forming a more robust feature geometry even in real-world data containing noisy or hard samples [4]. These studies suggest that when a class cannot be well described by a single distribution, sub-center-based representations can absorb intra-class variation and form a more stable decision boundary. This perspective is also suitable for AI-generated image detection: real images involve diverse acquisition sources and content distributions, while fake images exhibit different artifact modes depending on generator architecture, sampling strategy, and post-processing. Therefore, if we can reduce the dependence of CLIP image embeddings on caption-level semantics and map the resulting features to multiple sub-centers, we may improve the generalization of real/fake authentication to diverse unseen generative models.

To this end, we propose SAGE: Semantic-Anchor-Guided Embedding. SAGE learns representations with reduced dependence on caption-level semantics for real/fake discrimination through caption-guided residualization, projection, and decorrelation. To achieve this, we construct paired triplets consisting of COCO images (i.e., real), COCO images reconstructed by SDXL (i.e., fake), and COCO captions (i.e., text), so that real and fake images share the same caption-level semantics. This paired dataset encourages the model to focus on generation traces rather than semantic content. In addition, considering the realistic scenario where captions are unavailable at test time, we learn a semantic anchor, a learnable surrogate vector in the CLIP text feature space that approximates the semantic direction provided by paired text embeddings. For the final authentication stage, we adopt a SoftTriple-style sub-center classifier, considering that the diverse sources and artifact patterns within real/fake classes are difficult to represent with a single center. By representing each class with multiple sub-center proxies, this classifier models diverse local clusters and learns metric-like

embeddings without triplet sampling, thereby providing more stable authentication performance on unseen generators.

The contributions of this paper are summarized as follows: (**I**) We propose SAGE, a framework for learning feature representations with reduced dependence on caption-level semantics for AI-generated image detection. SAGE leverages an image–text multimodal structure and uses a novel paired dataset that reuses semantic information (i.e., captions) from an existing real-image dataset. (**II**) We introduce a sub-center-based learning strategy for AI-generated image detection. Designed for test-time scenarios involving diverse sources, this strategy achieves lower performance variation across different generators compared with existing detectors.

2 Related Works

AI-generated image detection. AI-generated image detection is defined as a binary authentication problem that distinguishes real images from generated images, and early studies focused on low-level artifacts and abnormal frequency patterns left by generative models. Wang et al. showed that a detector trained on ProGAN can generalize to various CNN-based generators such as StyleGAN, BigGAN, and CycleGAN, suggesting the possibility of exploiting systematic flaws shared by CNN-generated images [36]. Subsequent studies have sought to improve generalization performance by leveraging diverse forensic cues, including global-local feature fusion, gradient representations, frequency-domain cues, and up-sampling artifacts [12, 32–34]. For example, Ju et al. combined global features from the entire image with local features from informative patches, LGrad used gradients from a pretrained CNN as artifact representations, and FreqNet and NPR enhanced generalization to unseen sources by analyzing high-frequency information and neighboring pixel relationships left by up-sampling operations, respectively. More recently, methods based on large-scale pretrained models, especially CLIP, have attracted attention for their strong generalization ability. Ojha et al. showed that a pretrained vision-language feature space, which is not directly trained for real/fake classification, can provide better generalization than conventional supervised detectors, while Cozzolino et al. reported that a lightweight detector using CLIP features can achieve strong robustness to out-of-distribution (OOD) generators and post-processing even with limited training data [21, 3]. Meanwhile, FatFormer pointed out that the fixed paradigm of frozen CLIP with a linear classifier may be insufficient for learning forgery representations, and combined forgery-aware adaptation in the image/frequency domains with language-guided alignment [19]. These lines of work suggest that generative artifacts can appear in various forms, including spatial, frequency, and structural relations, and that although CLIP features provide useful foundation representations, task-specific adaptation is still required for robust AI-generated image detection.

In AI-generated image detection, domain generalization refers to the ability of a detector to operate reliably on generators, image sources, and sampling strategies that are not seen during training. Existing detectors are prone to overfitting to artifacts from specific generators or dataset-specific biases in the training data, and tend to misclassify fake images as real when encountering unseen generators. Ojha et al. explained this phenomenon through asymmetric tuning and the real sink-class effect, while Effort showed that naive detectors overfit to limited fake patterns and collapse the feature space into a constrained low-rank subspace, proposing orthogonal subspace decomposition to leverage the high-rank knowledge of foundation models [21, 40]. In addition, B-Free pointed out that detectors can learn dataset biases such as content, format, and resolution instead of generator-specific artifacts, and proposed a bias-free training paradigm that ensures semantic alignment between real and fake images. VIB-Net observed that authentication-related information can be buried under task-irrelevant information in pretrained multimodal features, and enforced task-relevant feature learning through a variational information bottleneck [9, 42].

3 Method

This section describes the overall architecture and training algorithm of the proposed SAGE. Figure 2 illustrates the decomposition process for semantic-anchor-guided representation learning and the overall training flow.

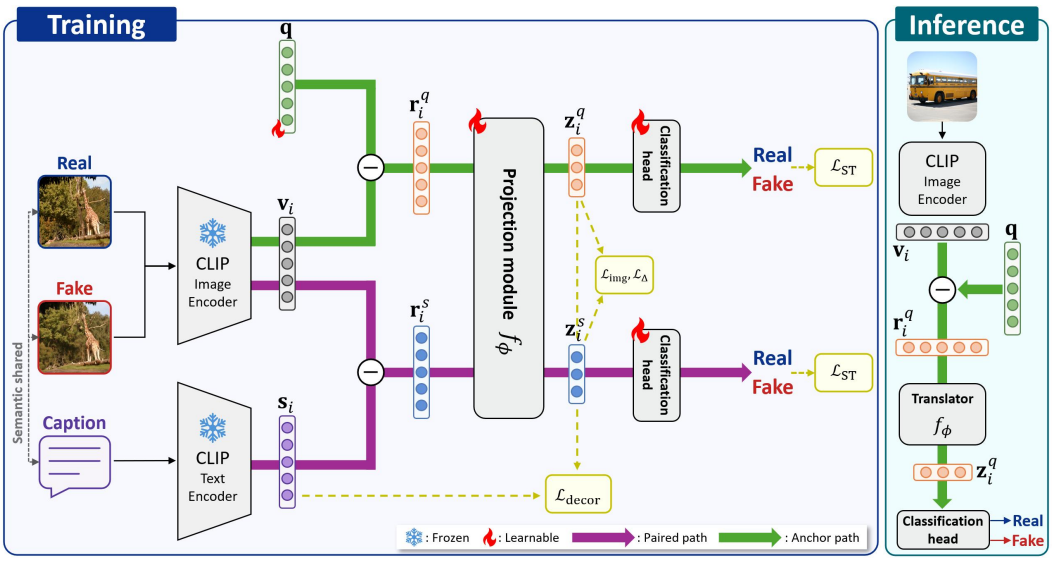

Figure 2: Framework overview of SAGE. During training, the paired text path forms caption-guided residuals using paired CLIP text embeddings, while the semantic anchor path learns a caption-free decomposition using a learnable surrogate anchor. At inference time, only the anchor path is used, enabling real/fake authentication without captions. L2 normalization is applied to each residual before projection and to each projected output.

3.1 Semantically Paired Authentication Dataset

Previous approaches have attempted to construct training datasets with real–fake pairs that share semantics by generating images from text prompts. [30, 10]. However, such dataset construction does not necessarily achieve strict visual-semantic alignment, since a single caption can produce images with diverse visual variations. Our goal is to construct a dataset that contains real and fake images of the same scene while sharing the same semantic content. By doing so, we encourage the detector to focus on generation traces rather than the semantic content of the image. To this end, we use the COCO dataset, which provides well-annotated captions for each image [18]. Specifically, we use COCO images as the real set and reconstruct each COCO image using SDXL [24] to construct the corresponding fake set.

3.2 SAGE: Semantic-Anchor-Guided Embedding

The training process of SAGE consists of two paths: the paired text decomposition path and the semantic anchor decomposition path (see Figure 2). The paired path forms a caption-guided residual from the image feature using the ground-truth caption embedding during training, whereas the anchor path is learned for caption-free inference at test time. Since captions are not available during inference, only the semantic anchor decomposition path is used.

Paired text decomposition. To provide a caption-guided starting point for reducing semantic dependence,, we use the residual between the CLIP image feature $\mathbf{v}_i$ and the CLIP text feature $\mathbf{s}_i$. Specifically, in the paired path, we compute $\mathbf{r}_i^s = \mathbf{v}_i - \mathbf{s}_i$ to obtain a caption-guided residual representation. The residual feature is L2-normalized before being passed through the projection module f_ϕ, and the output is normalized to obtain $\mathbf{z}_i^s = \mathrm{norm}(f_\phi(\mathrm{norm}(\mathbf{r}_i^s)))$, where norm denotes L2 normalization.

The image–text residual alone does not guarantee a fully semantic-free representation. Rather, it provides a semantically guided starting point, and the projection module f_ϕ, trained with classification and HSIC-based decorrelation, further transforms this residual into an authentication-oriented embedding. This interpretation is supported by our stage-wise analysis in Appendix E.3, where the largest reduction in CKA and dCor occurs after projection. Thus, f_ϕ compensates for the limitation

of a simple residual by extracting a compact representation that is more favorable for real/fake discrimination.

Semantic anchor decomposition. In the anchor path, we use the normalized semantic anchor $\mathbf{q}$ as a substitute for the text feature and compute $\mathbf{r}_i^q = \mathbf{v}_i - \mathrm{norm}(\mathbf{q})$. Then, using the same projection module as the paired path, we obtain $\mathbf{z}_i^q = \mathrm{norm}(f_\phi(\mathrm{norm}(\mathbf{r}_i^q)))$. Here, $\mathbf{z}_i^q$ has the same embedding dimension as $\mathbf{z}_i^s$, and the two paths share the same classifier and are coupled through consistency objectives. This allows the anchor path to learn the semantic decomposition effect provided by the paired text path, while enabling caption-free real/fake authentication at inference time. The contribution of the learned semantic anchor $\mathbf{q}$ is further analyzed in Appendix E.

The anchor is a global task-adaptive reference, not an image-specific semantic estimator. Post-subtraction normalization breaks exact fixed-bias equivalence, but the anchor transformation remains invertible for unit-norm inputs with nonzero residuals. Reduced semantic dependence is therefore attributed to the learned projection and joint objectives rather than anchor subtraction alone.

Training of the semantic anchor. Based on the above conditions, we aim to achieve two training objectives. First, the representation from the anchor path should have a discriminative structure similar to that of the paired path, so that the semantic anchor $\mathbf{q}$ can substitute for the paired text feature $\mathbf{s}$. Second, the projection module f_ϕ should preserve information useful for real/fake authentication while reducing semantic dependence in the decomposed feature. For the first objective, we train $\mathbf{q}$ using a sample-level caption–anchor consistency loss $\mathcal{L}_{\mathrm{img}}$. The loss is defined as follows:

$$\mathcal{L}_{\mathrm{img}} = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{z}_i^s - \mathbf{z}_i^q\|_2^2, \tag{1}$$

where N denotes the batch size. This loss encourages the paired path and the anchor path to produce similar decomposed features for the same image.

However, sample-level consistency alone does not guarantee that the two paths preserve the same real/fake separation structure. For example, even if each anchor feature is close to its corresponding paired feature, the class-mean difference, i.e., the real/fake separation direction, can differ between the two paths when the errors of real and fake samples accumulate in different directions [38, 37]. To alleviate this issue, we additionally introduce $\mathcal{L}_\Delta$to regularize the real/fake class-mean differences computed from the paired and anchor paths to be consistent, alongside $\mathcal{L}_{\mathrm{img}}$. Specifically, Δ is defined as follows:

$$\Delta = \mu_{\mathrm{fake}} - \mu_{\mathrm{real}}, \quad \text{where} \quad \mu_c = \frac{1}{|\mathcal{B}_c|}\sum_{i\in\mathcal{B}_c} z_i. \tag{2}$$

Here, $\mathcal{B}_c$ denotes the set of samples belonging to class c within the mini-batch, where c corresponds to either real or fake. The delta consistency loss using the two delta vectors is then defined as follows:

$$\mathcal{L}_\Delta = \|\Delta^s - \Delta^q\|_2^2. \tag{3}$$

This term encourages the anchor path to preserve not only the feature-level correspondence to the paired path, but also the real–vs–fake separation direction [38, 25]. As a result, the anchor path used at test time is trained to have an authentication geometry similar to that of the paired text path. A detailed analysis of $\mathcal{L}_\Delta$ is provided in Appendix B.

Semantic decorrelation learning. For the second objective, we weaken the dependence between the decomposed feature $\mathbf{z}_i^q$ and the semantic feature $\mathbf{s}_i$, thereby discouraging the classifier from exploiting semantic shortcuts that are irrelevant to real/fake discrimination. To this end, we introduce a semantic decorrelation loss that reduces the statistical dependence between the final representation of the anchor path (i.e., $\mathbf{z}_i^q$) and the paired text feature (i.e., $\mathbf{s}_i$). Specifically, we use the Hilbert-Schmidt Independence Criterion (HSIC) [7] to measure the dependence between $\mathbf{Z}^q = \{\mathbf{z}_i^q\}_{i=1}^N$ and $\mathbf{S} = \{\mathbf{s}_i\}_{i=1}^N$. A smaller HSIC value indicates lower statistical dependence between the two representations. In particular, when a linear kernel is used, minimizing HSIC penalizes the cross-covariance between batch-centered features [7, 8]. Accordingly, the decorrelation loss $\mathcal{L}_{\mathrm{decor}}$ is defined as follows:

$$\mathcal{L}_{\mathrm{decor}} = \frac{1}{N-1}\left\|\tilde{\mathbf{Z}}^{q\top}\tilde{\mathbf{S}}\right\|_F^2, \qquad \text{where} \quad \begin{cases}\tilde{\mathbf{Z}}^q = \mathbf{Z}^q - \mathbf{1}\mu_{\mathbf{Z}}^\top \\ \tilde{\mathbf{S}} = \mathbf{S} - \mathbf{1}\mu_{\mathbf{S}}^\top\end{cases}, \tag{4}$$

where N denotes the batch size, and $\|\cdot\|_F$ denotes the Frobenius norm. This loss penalizes cross-covariance between the anchor-path embeddings and the paired caption features, encouraging the final feature to focus more on forensic cues necessary for real/fake authentication rather than semantic content. A detailed derivation of $\mathcal{L}_{\text{decor}}$ is provided in Appendix C.

3.3 Sub-Center Authentication with SoftTriple

Fake images can exhibit different artifact patterns depending on generator architecture, training data, and sampling strategy, while real images also involve diverse sources and content distributions. Therefore, a classifier that represents each class with a single prototype may form an unstable decision boundary under unseen generator shifts. Although the preceding embedding exhibits reduced dependence on caption-level semantics, it does not fully address the fact that real/fake classes are still difficult to represent as single-mode distributions. Inspired by the SoftTriple metric learning framework proposed by Qian et al. [26], we adopt a sub-center classifier to mitigate this issue. Figure 3 provides a conceptual illustration of the applied SoftTriple loss.

Revisiting SoftTriple loss. In SoftTriple-style metric learning, each class c is represented not by a single center, but by K learnable sub-center proxies $\{\mathbf{w}_{c,k}\}_{k=1}^{K}$. In other words, each class c has K proxy vectors, and we compute the cosine similarity between the decomposed feature $\mathbf{z}$ and each proxy vector as follows:

$$\alpha_{c,k} = \frac{\mathbf{z}_i^\top \mathbf{w}_{c,k}}{\|\mathbf{z}_i\|\|\mathbf{w}_{c,k}\|}. \tag{5}$$

These proxy vectors serve as sub-centers within class c, providing reference points for measuring how similar the current input $\mathbf{z}$ is to each sub-center $\mathbf{w}_{c,k}$. The similarities $\boldsymbol{\alpha}_c = \{\alpha_{c,k}\}_{k=1}^{K}$ are computed separately for each class, and a class-wise softmax is applied over the K sub-centers to obtain the sub-center assignment weight $\omega_{c,k}$:

$$\omega_{c,k} = \frac{\exp(\alpha_{c,k}/\gamma)}{\sum_{j=1}^{K}\exp(\alpha_{c,j}/\gamma)} \tag{6}$$

where γ denotes the proxy temperature; a smaller γ assigns a larger weight to the nearest sub-center. Therefore, when a class contains multiple local modes, each sample can be represented near the sub-center that best matches its feature. The final logit l_c is then constructed as the class-wise softmax-weighted similarity, and the SoftTriple loss $\mathcal{L}_{\text{ST}}$ is defined using cross-entropy:

$$\mathcal{L}_{\text{ST}} = -\log\frac{\exp(\lambda l_y)}{\sum_{c=0}^{C-1}\exp(\lambda l_c)}, \qquad \text{where} \quad l_c = \sum_{k=1}^{K}\omega_{c,k}\cdot\alpha_{c,k}. \tag{7}$$

Here, y and C denote the class label and the number of classes, respectively, and λ is the logit scaling factor.

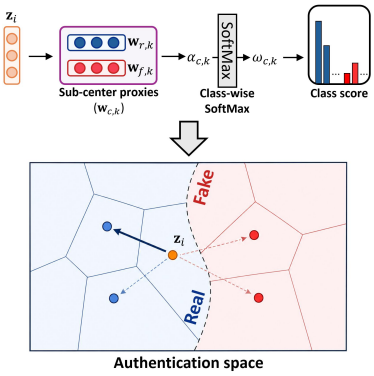

Figure 3: Conceptual illustration of SoftTriple loss for the authentication task. Each real/fake class is represented by multiple learnable sub-center proxies rather than a single class prototype.

As a result, this sub-center formulation can more flexibly model intra-class variation within real and fake classes by representing each class with multiple local modes rather than a single center. Consequently, SAGE learns a more stable authentication geometry across diverse unseen generators, instead of relying on a single decision boundary that may overfit to specific generator artifacts.

4 Experiments

In this section, we experimentally validate the performance of SAGE and the effectiveness of its key components. Our experiments aim to answer three questions. First, *does SAGE achieve more stable detection performance than existing methods across diverse unseen generators?* Second, *do the proposed semantic decomposition and decorrelation actually reduce semantic dependence and form a feature geometry favorable for real/fake authentication?* Third, *does the SoftTriple sub-center classifier contribute to learning a stable authentication geometry across diverse generator sources?*

Table 1: Cross-generator detection performance on representative CommunityForensics generators. We show six representative generators selected to cover commercial generators, open diffusion models, and challenging low-performing cases; the Avg. columns are computed over all 13 unseen generators listed in Appendix E.1. For each generator, we report Acc. / AUC.

Method	DALL·E 3		FLUX-dev		Firefly		Ideogram		Imagen3		MidJV5.2		Avg.		
	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	mAP
B-Free	84.5	92.3	68.2	75.8	72.3	76.3	58.8	59.7	52.3	46.4	57.9	83.8	80.9	85.8	87.3
CNNSpot	51.2	51.0	50.3	40.0	52.8	81.0	50.6	43.3	50.5	50.9	51.4	46.6	52.8	46.4	49.7
FreqNet	78.0	97.8	73.4	95.9	50.8	87.3	78.2	97.2	89.5	99.1	81.6	97.7	70.7	83.7	81.4
NPR	62.8	85.1	63.4	90.9	56.7	55.8	62.6	81.8	62.5	89.6	62.8	88.1	58.5	73.1	69.0
UniFD	57.7	75.1	47.8	38.5	81.5	93.8	52.3	66.1	51.1	59.1	54.0	62.8	56.3	57.7	58.4
VIBNet	93.0	99.2	80.9	88.5	56.9	71.4	89.8	95.8	88.5	95.8	78.4	84.5	77.1	87.0	84.8
Effort	90.5	96.8	87.2	94.1	59.1	68.1	90.1	96.5	88.1	95.1	85.6	92.7	74.9	90.1	88.6
Ours	86.7	94.5	83.2	90.8	82.5	90.3	83.4	90.0	80.8	90.2	87.6	94.8	**84.9**	**92.3**	**90.3**

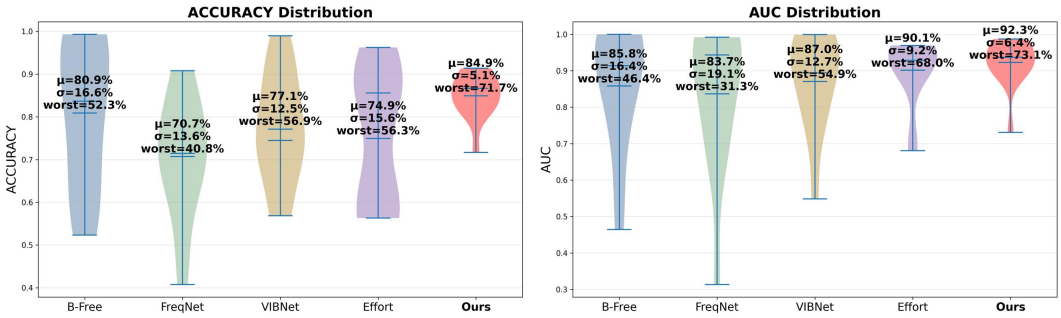

Figure 4: Generator-wise performance distributions of representative methods. We visualize both accuracy and AUC across unseen generators. μ and σ denote the average and standard deviation, respectively.

4.1 Experimental Settings

Evaluation datasets. We adopt CommunityForensics [22], MS-COCOAI [30], and WildRF [2] as evaluation datasets, with the main evaluation conducted on the CommunityForensics benchmark. CommunityForensics includes recent commercial and open text-to-image generators such as DALL·E 3, Firefly, FLUX, Ideogram, Imagen3, and Midjourney, making it suitable for evaluating unseen generator generalization in real-world scenarios. All generators are treated as unseen sources that are not used during training. We evaluate each generator separately and report both the generator-wise mean and standard deviation.

Baselines. We compare SAGE with CNNSpot [36], FreqNet [33], NPR [34], UniFD [21], VIBNet [42], Effort [40], and B-Free [9]. CNNSpot is a representative detector that exploits artifacts from CNN-generated images, while FreqNet and NPR are based on frequency cues and up-sampling artifacts, respectively. B-Free, UniFD, VIBNet, and Effort are strong recent baselines that address foundation-model-based detection, semantic bias, or generalization, making them directly relevant comparison methods for SAGE.

Evaluation metrics. We use Accuracy, AUC, and mAP as evaluation metrics. In addition, to assess the alignment and dependence between SAGE representations and paired CLIP text features, we adopt centered kernel alignment (CKA) [15] and distance correlation (dCor) [31]. Detailed formulations of these two metrics are provided in Appendix D.1.

4.2 Cross-Generator Detection Performance

Table 1 summarizes the detection performance on 13 unseen generators from CommunityForensics. SAGE shows strong competitiveness in terms of average accuracy and AUC, while substantially reducing generator-wise performance variation. Specifically, SAGE achieves an average accuracy of 84.9%, an average AUC of 92.3%, and an mAP of 90.4%, demonstrating competitive performance compared with recent CLIP-based detectors. Notably, SAGE achieves the lowest generator-wise

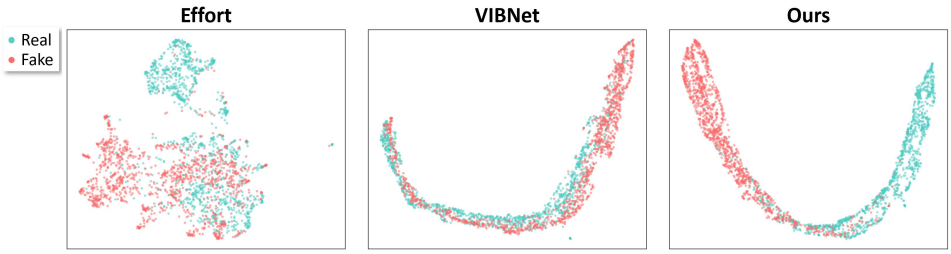

Figure 5: UMAP visualization of the final representations from Effort, VIBNet, and SAGE. We compare representative strong baselines that modify or regularize the feature space for generalizable detection.

standard deviation in accuracy, 5.1%, and the highest worst-generator accuracy, 71.7%. These results indicate that SAGE is not merely strong on a specific generator, but consistently suppresses performance degradation across diverse unseen generators.

Figure 4 visualizes the generator-wise accuracy and AUC distributions of representative methods. While several baselines achieve high performance on some generators, their distributions are wide and often include low-performance tails. In contrast, SAGE shows a relatively compact distribution with a high lower bound. This supports our main claim that semantic-anchor-guided embedding and sub-center authentication contribute not only to average performance but also to stability under generator shifts. Additional results on other evaluation datasets are provided in Appendix E.

4.3 Feature Geometry Analysis

Next, we analyze whether the learned representation forms a feature geometry more suitable for real/fake authentication. Figure 5 visualizes the final representations of Effort, VIBNet, and SAGE on the same CommunityForensics subset using UMAP. Effort shows partial overlap between real and fake samples, while VIBNet forms a smoother manifold but still contains regions where real and fake samples are mixed. In contrast, SAGE produces a clearer separation between real and fake clusters. This suggests that learning residual features with reduced semantic dependence leads to an embedding space that better reflects authentication-related signals.

4.4 Ablation Studies

Component ablation. Table 2 provides a component-wise ablation of SAGE on CommunityForensics. The image-only baselines indicate that simply adding an MLP or replacing BCE with SoftTriple is not enough to obtain the performance of SAGE. Although these variants improve over Raw CLIP, they still operate on CLIP image features in which semantic and authentication cues are entangled. This suggests that the key gain comes from explicitly constructing a semantically guided residual representation rather than from classifier capacity alone.

The SAGE variants show that each objective contributes to a different aspect of the learned representation. Without $\mathcal{L}_{\text{decor}}$, the anchor-side feature is less explicitly constrained to reduce its dependence on caption-level semantics, making the model more susceptible to semantic shortcuts. Without $\mathcal{L}_{\Delta}$,

Table 2: Component ablation on CommunityForensics. We isolate the contributions of semantic decomposition, HSIC-based semantic decorrelation, delta consistency, and SoftTriple sub-center authentication. For each variant, we report average Acc. / AUC over the selected unseen generators.

Variant	Decomp.	$\mathcal{L}_{\text{decor}}$	$\mathcal{L}_{\Delta}$	SoftTriple	Acc. / AUC
Raw CLIP + BCE	–	–	–	–	73.7 / 85.5
Image-only MLP + BCE	–	–	–	–	77.8 / 86.3
Image-only MLP + SoftTriple	–	–	–	✓	78.8 / 87.1
SAGE w/o $\mathcal{L}_{\text{decor}}$	✓	–	✓	✓	81.7 / 89.5
SAGE w/o $\mathcal{L}_{\Delta}$	✓	✓	–	✓	82.0 / 89.7
SAGE w/o SoftTriple	✓	✓	✓	–	81.5 / 91.5
SAGE	✓	✓	✓	✓	**84.9 / 92.3**

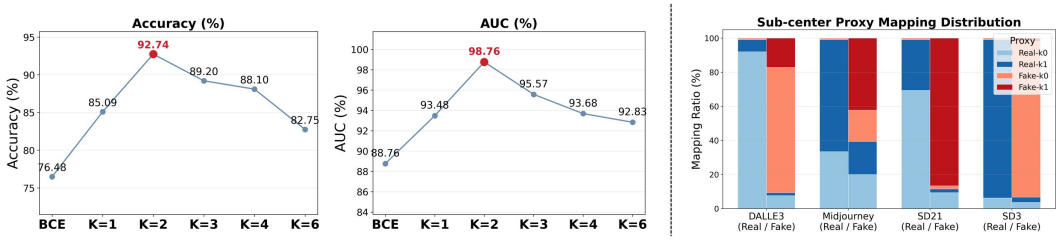

Figure 6: Ablation on the number of SoftTriple sub-centers K and the sub-center proxy mapping distribution. The mapping distribution shows that sub-centers remain class-aligned and provide multiple local anchors within each class. We interpret $K = 2$ as a balance between modeling intra-class variation and avoiding over-fragmentation, rather than as a one-to-one generator clustering.

the anchor path may match paired features at the sample level but fail to preserve the same real/fake separation direction, which weakens caption-free inference. Without SoftTriple, the model loses the ability to represent multiple local modes within each class, leading to a less stable decision boundary. These results show that SAGE is not a single-component improvement: semantic decomposition, decorrelation, delta consistency, and sub-center authentication jointly shape a representation that is less semantic-dominant and more aligned with real/fake authentication.

Semantic decorrelation analysis. We quantitatively evaluate whether SAGE actually reduces the semantic dependence of the final representation. To this end, we measure the dependence between feature embeddings and CLIP text embeddings using CKA and dCor. Lower CKA and dCor values indicate weaker measured alignment and dependence, respectively, between the learned features and the paired caption embeddings. As shown in Table 3, raw CLIP image features exhibit high alignment with text embeddings, with a CKA of 0.672 and a dCor of 0.892. In contrast, after semantic decomposition, CKA decreases to 0.051 and dCor decreases to 0.323, corresponding to reductions of 92.4% and 63.7%, respectively. These results support the core assumption of SAGE: the final representation exhibits reduced measured dependence on caption embeddings while retaining authentication-relevant information

Table 3: Semantic decorrelation analysis. We measure the dependence between image-side features and CLIP text embeddings. Lower CKA and dCor indicate weaker semantic dependence.

Feature	CKA ↓	dCor ↓
Raw CLIP image feature	0.672	0.892
Projected feature	0.051	0.323
Reduction	92.4%	63.7%

Effect of the number of sub-centers. We analyze the effect of the number of SoftTriple sub-centers K. This analysis is conducted on the MS-COCOAI dataset. Figure 6 shows the average accuracy and AUC as K varies. The BCE baseline, which does not use sub-center proxies, achieves 76.48% accuracy and 88.76% AUC. Using proxy-based angular classification with $K = 1$ already improves performance, suggesting that a normalized proxy classifier provides a stronger authentication geometry than standard BCE. The best performance is observed at $K = 2$, achieving 92.74% accuracy and 98.76% AUC. In contrast, increasing K beyond 2 leads to performance degradation. This indicates that an excessive number of sub-centers may over-fragment the representation space and hurt generalization performance.

The mapping distribution in Figure 6 further illustrates the role of the learned sub-centers. Real images are mapped mainly to real-class proxies, while fake images are mapped mainly to fake-class proxies, indicating that the sub-centers have learned class-aligned patterns. At the same time, different generators exhibit different preferences over the two sub-centers, suggesting that the classifier captures heterogeneous artifact modes rather than collapsing each class into a single prototype. This observation helps explain why an appropriate level of sub-center modeling enables stable authentication under unseen generator shifts.

5 Conclusion

This paper proposes SAGE, a semantic-anchor-guided embedding framework for generalized AI-generated image detection. SAGE constructs caption-guided residuals using CLIP image/text encoders and learns a semantic anchor for caption-free inference. The resulting residual features are refined by a projection module trained with path consistency, semantic decorrelation, and SoftTriple-style sub-center authentication. Experiments on CommunityForensics show that SAGE achieves competitive detection performance while substantially reducing accuracy variation across unseen generators and improving worst-case accuracy. Additional analyses confirm that SAGE reduces dependence on semantic text features and forms a clearer real/fake feature geometry than strong baselines. These results suggest that reducing dependence on caption-level semantics and modeling intra-class variation are important for stable AI-generated image detection under generator shifts.

6 Limitations and Societal Impact

Limitations. Although SAGE reduces measured dependence on caption-level semantics, it does not guarantee complete semantic independence. Its reliance on a frozen CLIP backbone, a single global semantic anchor, and COCO–SDXL training pairs may introduce representation and dataset-specific biases. Moreover, its lower performance on WildRF indicates remaining vulnerability to uncontrolled post-processing and domain shifts.

Societal impact. SAGE can support media forensics by improving detection across unseen generators. However, false positives may misclassify authentic images, while false negatives may allow synthetic misinformation to go undetected. Detector knowledge may also facilitate adversarial evasion. Therefore, SAGE should be used as a decision-support tool alongside provenance information and human review, rather than as sole evidence in high-stakes decisions. Further discussion is provided in Appendix F.

Acknowledgements

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (No. RS-2025-02263841, Development of a Realtime Multimodal Framework for Comprehensive Deepfake Detection Incorporating Common Sense Error Analysis, and No. RS-2025-02215344, Development of AI Technology with Robust and Flexible Resilience Against Risk Factors) and the Artificial Intelligence Industrial Convergence Cluster Development Project funded by the Ministry of Science and ICT (MSIT, Korea) and Gwangju Metropolitan City.

References

[1] G. Alain and Y. Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations*, 2017.

[2] B. Cavia, E. Horwitz, T. Reiss, and Y. Hoshen. Real-time deepfake detection in the real-world. *arXiv preprint arXiv:2406.09398*, 2024.

[3] D. Cozzolino, G. Poggi, R. Corvi, M. Nießner, and L. Verdoliva. Raising the bar of ai-generated image detection with clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4356–4366, 2024.

[4] J. Deng, J. Guo, T. Liu, M. Gong, and S. Zafeiriou. Sub-center arcface: Boosting face recognition by large-scale noisy web faces. In *European Conference on Computer Vision*, pages 741–757. Springer, 2020.

[5] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[6] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[7] A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In *Algorithmic Learning Theory*, pages 63–77. Springer, 2005. doi: 10.1007/11564089_7.

[8] A. Gretton, K. Fukumizu, C. H. Teo, L. Song, B. Schölkopf, and A. J. Smola. A kernel statistical test of independence. In *Advances in Neural Information Processing Systems*, volume 20, 2007.

[9] F. Guillaro, G. Zingarini, B. Usman, A. Sud, D. Cozzolino, and L. Verdoliva. A bias-free training paradigm for more general ai-generated image detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18685–18694, 2025.

[10] S. Gye, J. Ko, H. Shon, M. Kwon, and J. Kim. Sfld: Reducing the content bias for ai-generated image detection. *arXiv preprint arXiv:2502.17105*, 2025.

[11] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[12] Y. Ju, S. Jia, L. Ke, H. Xue, K. Nagano, and S. Lyu. Fusing global and local features for generalized ai-synthesized image detection. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 3465–3469. IEEE, 2022.

[13] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[14] M. Keita, W. Hamidouche, H. B. Eutamene, A. Taleb-Ahmed, and A. Hadid. Deeclip: A robust and generalizable transformer-based framework for detecting ai-generated images. *arXiv preprint arXiv:2504.19876*, 2025.

[15] S. Kornblith, M. Norouzi, H. Lee, and G. Hinton. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3519–3529, 2019.

[16] J. Li, D. Li, C. Xiong, and S. Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022.

[17] W. Liang, Y. Zhang, Y. Kwon, S. Yeung, and J. Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

[18] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[19] H. Liu, Z. Tan, C. Tan, Y. Wei, J. Wang, and Y. Zhao. Forgery-aware adaptive transformer for generalizable synthetic image detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10770–10780, 2024.

[20] F. Marra, D. Gragnaniello, L. Verdoliva, and G. Poggi. Do gans leave artificial fingerprints? In *2019 IEEE conference on multimedia information processing and retrieval (MIPR)*, pages 506–511. IEEE, 2019.

[21] U. Ojha, Y. Li, and Y. J. Lee. Towards universal fake image detectors that generalize across generative models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24480–24489, 2023.

[22] J. Park and A. Owens. Community forensics: Using thousands of generators to train fake image detectors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8245–8257, 2025.

[23] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu. Semantic image synthesis with spatially-adaptive normalization. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 2332–2341. IEEE, 2019.

[24] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[25] C. Qi and F. Su. Contrastive-center loss for deep neural networks. In *2017 IEEE International Conference on Image Processing*, pages 2851–2855. IEEE, 2017. doi: 10.1109/ICIP.2017.8296803.

[26] Q. Qian, L. Shang, B. Sun, J. Hu, H. Li, and R. Jin. Softtriple loss: Deep metric learning without triplet sampling. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6450–6458, 2019.

[27] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[28] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021.

[29] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[30] R. Roy, N. Imanpour, A. Aziz, S. Bajpai, G. Singh, S. Biswas, K. Wanaskar, P. Patwa, S. Ghosh, S. Dixit, et al. A comprehensive dataset for human vs. ai generated image detection. *arXiv preprint arXiv:2601.00553*, 2026.

[31] G. J. Székely, M. L. Rizzo, and N. K. Bakirov. Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, 35(6):2769–2794, 2007. doi: 10.1214/009053607000000505.

[32] C. Tan, Y. Zhao, S. Wei, G. Gu, and Y. Wei. Learning on gradients: Generalized artifacts representation for gan-generated images detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12105–12114, 2023.

[33] C. Tan, Y. Zhao, S. Wei, G. Gu, P. Liu, and Y. Wei. Frequency-aware deepfake detection: Improving generalizability through frequency space domain learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 5052–5060, 2024.

[34] C. Tan, Y. Zhao, S. Wei, G. Gu, P. Liu, and Y. Wei. Rethinking the up-sampling operations in cnn-based generative network for generalizable deepfake detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 28130–28139, 2024.

[35] A. Van Den Oord, N. Kalchbrenner, and K. Kavukcuoglu. Pixel recurrent neural networks. In *International conference on machine learning*, pages 1747–1756. PMLR, 2016.

[36] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros. Cnn-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8695–8704, 2020.

[37] T. Wang and P. Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020.

[38] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, pages 499–515. Springer, 2016.

[39] S. Yan, O. Li, J. Cai, Y. Hao, X. Jiang, Y. Hu, and W. Xie. A sanity check for ai-generated image detection. *arXiv preprint arXiv:2406.19435*, 2024.

[40] Z. Yan, J. Wang, P. Jin, K.-Y. Zhang, C. Liu, S. Chen, T. Yao, S. Ding, B. Wu, and L. Yuan. Orthogonal subspace decomposition for generalizable ai-generated image detection. *arXiv preprint arXiv:2411.15633*, 2024.

[41] N. Yu, L. S. Davis, and M. Fritz. Attributing fake images to gans: Learning and analyzing gan fingerprints. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7556–7566, 2019.

[42] H. Zhang, Q. He, X. Bi, W. Li, B. Liu, and B. Xiao. Towards universal ai-generated image detection by variational information bottleneck network. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 23828–23837, 2025.

Table 4: Summary of notation used in SAGE.

Notation	Description
$y \in \{0, 1\}$	Binary label, where 0 denotes real and 1 denotes fake.
N	Mini-batch size.
C	Number of classes. In our task, $C = 2$.
K	Number of SoftTriple sub-center proxies per class.
$E_I(\cdot)$	Frozen CLIP image encoder.
$E_T(\cdot)$	Frozen CLIP text encoder.
$\mathbf{v}_i \in \mathbb{R}^d$	CLIP image feature, $\mathbf{v}_i = \mathrm{norm}(E_I(x_i))$.
$\mathbf{s}_i \in \mathbb{R}^d$	CLIP text feature, $\mathbf{s}_i = \mathrm{norm}(E_T(c_i))$.
$\mathbf{q} \in \mathbb{R}^d$	Learnable semantic anchor used as a surrogate for text features at inference time.
$\mathbf{r}_i^s$	Paired text residual, $\mathbf{r}_i^s = \mathbf{v}_i - \mathbf{s}_i$.
$\mathbf{r}_i^q$	Semantic anchor residual, $\mathbf{r}_i^q = \mathbf{v}_i - \mathrm{norm}(\mathbf{q})$.
$f_\phi(\cdot)$	Projection module that maps residual features to compact embeddings.
$\mathbf{z}_i^s \in \mathbb{R}^m$	Projected embedding from the paired text path, $\mathbf{z}_i^s = \mathrm{norm}(f_\phi(\mathrm{norm}(\mathbf{r}_i^s)))$.
$\mathbf{z}_i^q \in \mathbb{R}^m$	Projected embedding from the semantic anchor path, $\mathbf{z}_i^q = \mathrm{norm}(f_\phi(\mathrm{norm}(\mathbf{r}_i^q)))$.
$\mathbf{Z}^q \in \mathbb{R}^{N \times m}$	Mini-batch matrix of anchor path embeddings.
$\mathbf{S} \in \mathbb{R}^{N \times d}$	Mini-batch matrix of CLIP text features.
$\tilde{\mathbf{Z}}^q$	Batch-centered anchor embedding matrix.
$\tilde{\mathbf{S}}$	Batch-centered semantic text feature matrix.
$\mu^s_{\mathrm{real}}, \mu^s_{\mathrm{fake}}$	Real/fake class means in the paired path.
$\mu^q_{\mathrm{real}}, \mu^q_{\mathrm{fake}}$	Real/fake class means in the anchor path.
Δ^s	Real/fake separation vector in the paired path, $\Delta^s = \mu^s_{\mathrm{fake}} - \mu^s_{\mathrm{real}}$.
Δ^q	Real/fake separation vector in the anchor path, $\Delta^q = \mu^q_{\mathrm{fake}} - \mu^q_{\mathrm{real}}$.
$\mathbf{w}_{c,k}$	k-th SoftTriple sub-center proxy for class c.
$\alpha_{c,k}$	Cosine similarity between $\mathbf{z}$ and $\mathbf{w}_{c,k}$.
$\omega_{c,k}$	Soft assignment weight for the k-th sub-center of class c.
l_c	Class-level SoftTriple similarity/logit before scaling.
γ	Temperature controlling sub-center assignment sharpness.
λ	Logit scaling factor in the SoftTriple classifier.
$\mathcal{L}_{\mathrm{ST}}$	SoftTriple classification loss.
$\mathcal{L}_{\mathrm{img}}$	Image-level consistency loss between $\mathbf{z}^s$ and $\mathbf{z}^q$.
$\mathcal{L}_{\Delta}$	Class-delta consistency loss between Δ^s and Δ^q.
$\mathcal{L}_{\mathrm{decor}}$	HSIC-based semantic decorrelation loss between $\mathbf{Z}^q$ and $\mathbf{S}$.
$\mathcal{L}_{\mathrm{SAGE}}$	Overall training objective of SAGE.

A Additional Details on Notation and Training Algorithm

This section summarizes the main notation and training algorithm of SAGE. SAGE uses frozen CLIP image and text encoders [27], a SoftTriple-style sub-center classifier [26], and HSIC-based semantic decorrelation [7, 8]. For readability, some equations are presented in a simplified form in the main text. SAGE jointly trains the paired text decomposition path and the semantic anchor decomposition path, and its final objective optimizes classification, path consistency, and semantic decorrelation together. Table 4 summarizes the main notation used in this paper, and Algorithm 1 presents the overall training procedure, including HSIC-based decorrelation.

A.1 Notation

Table 4 summarizes the notation used throughout the paper. We denote vectors by bold lowercase letters, matrices by bold uppercase letters, and scalar quantities by italic letters. The superscript s indicates the paired text path, while q indicates the semantic anchor path.

Algorithm 1 Training procedure of SAGE

Require: Training set $\mathcal{D} = \{(x_i, c_i, y_i)\}$, frozen CLIP encoders E_I, E_T
Require: Semantic anchor $\mathbf{q}$, projection module f_ϕ, SoftTriple head h_ψ
Require: Loss weights $\lambda_{\text{img}}, \lambda_\Delta, \beta$
Ensure: Trained parameters $\{\mathbf{q}, \phi, \psi\}$

for each mini-batch $\mathcal{B} = \{(x_i, c_i, y_i)\}_{i=1}^N$ **do**
  $\mathbf{v}_i \leftarrow \text{norm}(E_I(x_i)), \quad \mathbf{s}_i \leftarrow \text{norm}(E_T(c_i))$
  $\mathbf{r}_i^s \leftarrow \mathbf{v}_i - \mathbf{s}_i, \quad \mathbf{r}_i^q \leftarrow \mathbf{v}_i - \text{norm}(\mathbf{q})$
  $\mathbf{z}_i^s \leftarrow \text{norm}(f_\phi(\text{norm}(\mathbf{r}_i^s))), \quad \mathbf{z}_i^q \leftarrow \text{norm}(f_\phi(\text{norm}(\mathbf{r}_i^q)))$
  $\mathcal{L}_{\text{ce}} \leftarrow \mathcal{L}_{\text{ST}}(\mathbf{z}_i^s, y) + \mathcal{L}_{\text{ST}}(\mathbf{z}_i^q, y)$
  $\mathcal{L}_{\text{img}} \leftarrow \frac{1}{N}\sum_{i=1}^N \|\mathbf{z}_i^s - \mathbf{z}_i^q\|_2^2$
  $\Delta^s \leftarrow \mu_{\text{fake}}^s - \mu_{\text{real}}^s, \quad \Delta^q \leftarrow \mu_{\text{fake}}^q - \mu_{\text{real}}^q$
  $\mathcal{L}_\Delta \leftarrow \|\Delta^s - \Delta^q\|_2^2$
  $\mathcal{L}_{\text{decor}} \leftarrow \text{HSIC}_{\text{lin}}(\mathbf{Z}^q, \mathbf{S})$
  $\mathcal{L} \leftarrow \mathcal{L}_{\text{ce}} + \lambda_{\text{img}}\mathcal{L}_{\text{img}} + \lambda_\Delta\mathcal{L}_\Delta + \beta\mathcal{L}_{\text{decor}}$
  Update $\{\mathbf{q}, \phi, \psi\}$ by minimizing $\mathcal{L}$
end for

A.2 Training Algorithm

Algorithm 1 describes the training procedure of SAGE. During training, both paired text decomposition and semantic anchor decomposition are used. The paired path provides caption-guided semantic decomposition, while the anchor path learns to approximate this decomposition without requiring captions. The classifier and projection module are shared across the two paths. At inference time, only the semantic anchor path is used.

B Why Delta Consistency is Needed

In SAGE, the anchor path is trained to approximate the paired text decomposition path, enabling caption-free inference. A straightforward way to achieve this is to minimize a sample-level consistency loss between the paired feature $\mathbf{z}_i^s$ and the anchor feature $\mathbf{z}_i^q$. However, sample-level consistency alone does not guarantee that the two paths preserve the same class-level real/fake separation geometry.

To see this, we define the feature discrepancy between the anchor path and the paired path for sample i as follows:

$$\mathbf{e}_i = \mathbf{z}_i^q - \mathbf{z}_i^s. \tag{8}$$

Although $\mathcal{L}_{\text{img}}$ encourages each $\mathbf{e}_i$ to be small, the feature discrepancies can have different average directions for real and fake samples. We define the class-wise mean errors as follows:

$$\bar{\mathbf{e}}_{\text{real}} = \frac{1}{|\mathcal{B}_{\text{real}}|}\sum_{i\in\mathcal{B}_{\text{real}}}(\mathbf{z}_i^q - \mathbf{z}_i^s), \qquad \bar{\mathbf{e}}_{\text{fake}} = \frac{1}{|\mathcal{B}_{\text{fake}}|}\sum_{i\in\mathcal{B}_{\text{fake}}}(\mathbf{z}_i^q - \mathbf{z}_i^s), \tag{9}$$

The class means of the anchor path can then be written as:

$$\mu_{\text{real}}^q = \mu_{\text{real}}^s + \bar{\mathbf{e}}_{\text{real}}, \qquad \mu_{\text{fake}}^q = \mu_{\text{fake}}^s + \bar{\mathbf{e}}_{\text{fake}}. \tag{10}$$

Therefore, the real/fake separation direction of the anchor path can be expressed as:

$$\begin{aligned}\Delta^q &= \mu_{\text{fake}}^q - \mu_{\text{real}}^q \\ &= \Delta^s + (\bar{\mathbf{e}}_{\text{fake}} - \bar{\mathbf{e}}_{\text{real}}),\end{aligned} \tag{11}$$

where $\Delta^s = \mu_{\text{fake}}^s - \mu_{\text{real}}^s$. This derivation shows that, even when the feature discrepancy of each individual sample is small, the real/fake separation direction of the anchor path can differ from that of the paired path if the class-wise mean errors are different. In other words, the anchor path may approximate the paired features well at the sample level, while still forming a different real/fake geometry at the class level.

To alleviate this issue, we introduce the delta consistency loss:

$$\mathcal{L}_\Delta = \|\Delta^s - \Delta^q\|_2^2. \tag{12}$$

This objective directly regularizes the anchor path to preserve the real/fake separation direction of the paired path. As a result, the anchor path used at inference time is trained to inherit not only the sample-level feature correspondence, but also the authentication-oriented class-level geometry from the paired text decomposition path.

C Relation between Linear HSIC and Cross-Covariance

This section provides a formal derivation of the semantic decorrelation objective used in SAGE based on HSIC [7, 8]. In the main text, we define $\mathcal{L}_{\text{decor}}$ as a batch-centered cross-covariance penalty between the anchor-side representation and the paired text embedding. Here, we show that this objective corresponds to the empirical HSIC with a linear kernel, up to a constant scaling factor. This derivation clarifies why minimizing $\mathcal{L}_{\text{decor}}$ reduces the linear statistical dependence between the learned image-side feature and the caption-level semantic feature.

Lemma 1. *Let $Z \in \mathbb{R}^{N\times d_z}$ and $S \in \mathbb{R}^{N\times d_s}$ be two mini-batch representations, and let $H = I_N - \frac{1}{N}\mathbf{1}\mathbf{1}^\top$ be the centering matrix. When linear kernels are used, the empirical HSIC between Z and S is proportional to the squared Frobenius norm of their batch cross-covariance:*

$$\text{HSIC}_{\text{lin}}(Z,S) = \frac{1}{(N-1)^2}\left\|(HZ)^\top(HS)\right\|_F^2 \propto \|\text{Cov}(Z,S)\|_F^2 .$$

Proof. The empirical HSIC is defined as

$$\text{HSIC}(Z,S) = \frac{1}{(N-1)^2}\text{tr}(KHLH),$$

where K and L are Gram matrices. With linear kernels,

$$K = ZZ^\top, \qquad L = SS^\top.$$

Let $\tilde{Z} = HZ$ and $\tilde{S} = HS$. Since H is symmetric and idempotent, we have

$$HZZ^\top H = \tilde{Z}\tilde{Z}^\top, \qquad HSS^\top H = \tilde{S}\tilde{S}^\top.$$

Therefore,

$$\begin{aligned}\text{HSIC}_{\text{lin}}(Z,S) &= \frac{1}{(N-1)^2}\text{tr}\left(\tilde{Z}\tilde{Z}^\top\tilde{S}\tilde{S}^\top\right)\\ &= \frac{1}{(N-1)^2}\text{tr}\left(\tilde{Z}^\top\tilde{S}\tilde{S}^\top\tilde{Z}\right)\\ &= \frac{1}{(N-1)^2}\left\|\tilde{Z}^\top\tilde{S}\right\|_F^2 .\end{aligned}$$

The batch cross-covariance is

$$\text{Cov}(Z,S) = \frac{1}{N-1}\tilde{Z}^\top\tilde{S}.$$

Thus,

$$\text{HSIC}_{\text{lin}}(Z,S) = \|\text{Cov}(Z,S)\|_F^2 .$$

This shows that minimizing linear HSIC is equivalent to minimizing the squared Frobenius norm of the cross-covariance between Z and S. □

Remark. We use the simplified scaling $\frac{1}{N-1}$ in the training objective, since constant normalization factors are absorbed into the loss weight β and do not affect the optimization direction.

D Experimental Settings

This section provides additional details on the evaluation metrics and implementation settings used in our experiments.

D.1 Metrics

In the main experiments, we report Accuracy, AUC, and mAP for real/fake authentication. Accuracy is computed using a fixed threshold of 0.5. AUC measures the threshold-independent ranking capability of the detector, and mAP denotes the mean average precision across generators. In addition to detection performance, we use representation-level metrics to analyze whether the learned feature exhibits reduced dependence on caption-level semantics while preserving authentication-relevant structure. Specifically, we measure semantic dependence between image-side features and CLIP text embeddings using centered kernel alignment (CKA) and distance correlation (dCor). We further evaluate authentication utility using linear probe accuracy and linear probe AUC, and quantify real/fake feature separability using the Fisher ratio and centroid distance.

Centered kernel alignment. CKA measures the similarity between two representation spaces based on their centered Gram matrices [15]. Given two mini-batch representations $\mathbf{X} \in \mathbb{R}^{N \times d_x}$ and $\mathbf{Y} \in \mathbb{R}^{N \times d_y}$, we first compute their Gram matrices:

$$\mathbf{K} = \mathbf{X}\mathbf{X}^\top, \qquad \mathbf{L} = \mathbf{Y}\mathbf{Y}^\top. \tag{13}$$

Let $\mathbf{H} = \mathbf{I}_N - \frac{1}{N}\mathbf{1}\mathbf{1}^\top$ be the centering matrix. The centered Gram matrices are then given by

$$\tilde{\mathbf{K}} = \mathbf{H}\mathbf{K}\mathbf{H}, \qquad \tilde{\mathbf{L}} = \mathbf{H}\mathbf{L}\mathbf{H}. \tag{14}$$

The linear CKA score is defined as

$$\mathrm{CKA}(\mathbf{X}, \mathbf{Y}) = \frac{\langle \tilde{\mathbf{K}}, \tilde{\mathbf{L}} \rangle_F}{\|\tilde{\mathbf{K}}\|_F \|\tilde{\mathbf{L}}\|_F}, \tag{15}$$

where $\langle \cdot, \cdot \rangle_F$ denotes the Frobenius inner product. A larger CKA value indicates that the two representations have more similar sample-wise geometry. In our semantic decorrelation analysis, lower CKA between the learned image-side feature and the CLIP text embedding indicates lower similarity between their sample-wise geometries.

Distance correlation. Distance correlation measures statistical dependence between two sets of variables based on pairwise distances [31]. Given representations $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$ and $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^N$, we compute pairwise Euclidean distance matrices:

$$a_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2, \qquad b_{ij} = \|\mathbf{y}_i - \mathbf{y}_j\|_2. \tag{16}$$

The distance matrices are double-centered as

$$A_{ij} = a_{ij} - \bar{a}_{i\cdot} - \bar{a}_{\cdot j} + \bar{a}_{\cdot\cdot}, \qquad B_{ij} = b_{ij} - \bar{b}_{i\cdot} - \bar{b}_{\cdot j} + \bar{b}_{\cdot\cdot}, \tag{17}$$

where $\bar{a}_{i\cdot}$ and $\bar{a}_{\cdot j}$ denote row-wise and column-wise means, and $\bar{a}_{\cdot\cdot}$ denotes the global mean. The same notation applies to $\mathbf{B}$. The empirical distance covariance is

$$\mathrm{dCov}^2(\mathbf{X}, \mathbf{Y}) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} A_{ij} B_{ij}. \tag{18}$$

The distance correlation is then computed as

$$\mathrm{dCor}(\mathbf{X}, \mathbf{Y}) = \frac{\mathrm{dCov}(\mathbf{X}, \mathbf{Y})}{\sqrt{\mathrm{dCov}(\mathbf{X}, \mathbf{X})\mathrm{dCov}(\mathbf{Y}, \mathbf{Y})}}. \tag{19}$$

A smaller dCor value indicates weaker statistical dependence. Unlike CKA, dCor is based on pairwise distances and can capture more general dependence patterns. Therefore, we use both CKA and dCor to evaluate whether SAGE reduces semantic dependence from complementary perspectives.

Linear probe metrics. To evaluate whether each feature stage contains information useful for real/fake authentication, we train a lightweight linear probe on top of frozen representations [1] and report Probe Acc. and Probe AUC. Probe Acc. measures the classification accuracy at a fixed threshold, while Probe AUC measures the threshold-independent ranking ability of the probe. Higher values indicate that the representation is more informative for real/fake discrimination.

Feature separability metrics. We further quantify real/fake separability using the Fisher ratio and centroid distance. Let $\boldsymbol{\mu}_{\text{real}}$ and $\boldsymbol{\mu}_{\text{fake}}$ denote the class centroids, and let σ^2_{real} and σ^2_{fake} denote the intra-class variances. The Fisher ratio is defined as

$$\text{Fisher} = \frac{\|\mu_{\text{fake}} - \mu_{\text{real}}\|_2^2}{\sigma^2_{\text{real}} + \sigma^2_{\text{fake}}}. \tag{20}$$

A higher Fisher ratio indicates larger inter-class separation relative to intra-class variation. The centroid distance is defined as

$$D_{\text{centroid}} = \|\mu_{\text{fake}} - \mu_{\text{real}}\|_2, \tag{21}$$

which directly measures the distance between the real and fake class means. We use these metrics to assess whether the learned representation forms a more authentication-oriented feature geometry.

D.2 Implementation Details

Backbone and feature extraction. We use CLIP ViT-L/14 as the frozen image–text backbone [27]. The CLIP image encoder E_I extracts image embeddings, and the CLIP text encoder E_T extracts caption embeddings. Both image and text embeddings are L2-normalized before the decomposition step. The CLIP encoders are kept frozen throughout training, and only the semantic anchor, projection module, and classifier head are optimized.

Training data construction. The training set consists of COCO images [18], SDXL-reconstructed COCO images [24], and COCO captions. COCO images are used as real samples, while fake samples are constructed by reconstructing the corresponding COCO images using SDXL. The caption of each original COCO image is reused for both its real and reconstructed fake counterpart. This construction ensures that real/fake pairs share the same caption-level semantics. The training set contains 50K real images and 50K fake images.

Projection module. The projection module f_ϕ maps the L2-normalized residual feature to a compact authentication embedding space. In our implementation, f_ϕ is an MLP with dimension $768 \to 1024 \to 1024 \to 256$. Each hidden layer is followed by LayerNorm, GELU activation, and Dropout. The output embedding is L2-normalized before being passed to the classifier. Unless otherwise stated, the final embedding dimension is 256.

Semantic anchor. The semantic anchor $\mathbf{q}$ is a learnable vector defined in the CLIP text feature space. It has the same dimension as the CLIP text embedding and is L2-normalized before being subtracted from the CLIP image feature. During training, the anchor path is optimized to approximate the paired text decomposition path through image-level consistency and class-delta consistency. During inference, captions are not required, and only the anchor path is used.

SoftTriple classifier. We use a SoftTriple-style sub-center classifier for real/fake authentication [26]. Each class has K learnable sub-center proxies. Unless otherwise specified, we set $K = 2$. The scale factor and proxy temperature are set to $\lambda = 20.0$ and $\gamma = 0.1$, respectively. All classifier proxies and input embeddings are L2-normalized before cosine similarity computation.

Loss weights and optimization. The overall training objective is

$$\mathcal{L}_{\text{SAGE}} = \mathcal{L}_{\text{ce}} + \lambda_{\text{img}}\mathcal{L}_{\text{img}} + \lambda_\Delta\mathcal{L}_\Delta + \beta\mathcal{L}_{\text{decor}}. \tag{22}$$

$$\mathcal{L}_{\text{ce}} = \mathcal{L}_{\text{ST}}(\mathbf{z}^s, y) + \mathcal{L}_{\text{ST}}(\mathbf{z}^q, y). \tag{23}$$

We use $\lambda_{\text{img}} = 1.0$, $\lambda_\Delta = 0.5$, and $\beta = 0.1$ in the main experiments. The model is trained using AdamW with learning rate $1\text{e}{-4}$, weight decay $1\text{e}{-2}$, batch size 256, and maximum epochs 50. The best checkpoint is selected based on validation performance. All experiments were conducted on 1 NVIDIA H200 GPU. Training the main SAGE model took approximately 8 hours.

Inference protocol. At inference time, SAGE operates without captions. Given an input image x_i, we compute its CLIP image feature $\mathbf{v}_i = \text{norm}(E_I(x_i))$ and subtract the normalized semantic anchor $\text{norm}(\mathbf{q})$ to obtain the anchor residual $\mathbf{r}_i^q = \mathbf{v}_i - \text{norm}(\mathbf{q})$. The residual is L2-normalized before projection, giving $\mathbf{z}_i^q = \text{norm}(f_\phi(\text{norm}(\mathbf{r}_i^q)))$, and the SoftTriple classifier outputs the real/fake prediction score. Accuracy is computed with a threshold of 0.5.

Table 5: Full CommunityForensics results on 13 unseen generators, part I. For each generator, we report Acc. and AUC. The full average over all generators is reported in Table 6.

Method	DFGAN		DALL·E 3		DeciDiff.		FLUX-dev		FLUX-sch.		Firefly		Hourglass	
	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC
B-Free	99.2	100.0	84.5	92.3	99.3	100.0	68.2	75.8	83.7	90.9	72.3	76.3	82.9	91.4
CNNSpot	70.0	96.4	51.2	51.0	47.3	26.0	50.3	40.0	50.2	38.5	52.8	81.0	66.6	49.3
FreqNet	90.8	99.2	78.0	97.8	40.8	31.3	73.4	95.9	71.0	94.3	50.8	87.3	66.4	73.4
NPR	76.9	90.1	62.8	85.1	47.5	32.8	63.4	90.9	63.6	90.2	56.7	55.8	51.0	69.0
UniFD	99.0	100.0	57.7	75.1	43.3	38.5	47.8	38.5	48.1	44.7	81.5	93.8	71.7	68.8
VIBNet	99.0	99.9	93.0	99.2	66.2	76.9	80.9	88.5	72.5	81.5	56.9	71.4	57.3	54.9
Effort	96.2	97.0	90.5	96.8	56.9	88.6	87.2	94.1	85.9	92.9	59.1	68.1	63.6	70.5
Ours	87.1	98.2	86.7	94.5	82.0	90.3	83.2	90.8	86.7	93.7	82.5	90.3	71.7	73.1

Table 6: Full CommunityForensics results on 13 unseen generators, part II. Avg. Acc., Avg. AUC, and mAP are computed over all 13 selected generators.

Method	Ideogram		Imagen3		LCM-LoRA		MidJV5.2		Kandinsky		StableCas.		Avg.		
	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	mAP
B-Free	58.8	59.7	52.3	46.4	99.1	100.0	57.9	83.8	98.2	99.9	95.2	99.5	80.9	85.8	87.3
CNNSpot	50.6	43.3	50.5	50.9	47.9	17.6	51.4	46.6	47.3	20.9	49.6	41.3	52.8	46.4	49.7
FreqNet	78.2	97.2	89.5	99.1	61.2	64.6	81.6	97.7	65.8	70.5	71.5	79.3	70.7	83.7	81.4
NPR	62.6	81.8	62.5	89.6	50.0	58.4	62.8	88.1	50.9	60.9	49.4	57.3	58.5	73.1	69.0
UniFD	52.3	66.1	51.1	59.1	34.5	18.1	54.0	62.8	40.6	31.2	50.8	53.4	56.3	57.7	58.4
VIBNet	89.8	95.8	88.5	95.8	71.9	89.4	78.4	84.5	74.5	97.6	73.8	96.0	77.1	87.0	84.8
Effort	90.1	96.5	88.1	95.1	57.2	92.5	85.6	92.7	56.3	95.6	57.1	91.4	74.9	90.1	88.6
Ours	83.4	90.0	80.8	90.2	91.1	98.0	87.6	94.8	89.7	97.1	91.3	98.7	**84.9**	**92.3**	**90.3**

Existing assets and licenses. We use publicly available datasets and pretrained models, including COCO, CLIP, SDXL, CommunityForensics, MS-COCOAI, and WildRF. These assets are used for research purposes, and the corresponding original works are cited throughout the paper. We follow the usage terms and licenses specified by the original asset providers. Our generated SDXL reconstructions are used only for training and evaluation in this research context and are not released as a standalone public dataset in this submission.

E Additional Experimental Results

This section presents supplementary experimental results that further support the effectiveness of SAGE. In addition to the main evaluation on CommunityForensics [22], we report cross-dataset results on MS-COCOAI [30] and WildRF [2], and analyze the contribution of the learned semantic anchor. We further provide representation-level evidence through stage-wise analysis, feature similarity heatmaps, and nearest-neighbor retrieval, showing that SAGE reduces semantic dependence and forms an authentication-oriented feature geometry.

E.1 Cross-Dataset Generalization

CommunityForensics generator selection. For the CommunityForensics evaluation, we consider 13 unseen generators after excluding multiple-version variants of the same source model family. This avoids over-representing a single generator family in the evaluation and provides a more balanced estimate of cross-generator generalization. The selected generators are DFGAN, DALL·E 3, DeciDiffusionV2, FLUX-dev, FLUX-schnell, Firefly, Hourglass, IdeogramV1, Imagen3, LCM-LoRA-SDv1.5, MidjourneyV5.2, Kandinsky-2.2, and Stable Cascade. All of these generators are treated as unseen sources because SAGE is trained only on COCO images and SDXL-reconstructed COCO images. Therefore, the CommunityForensics evaluation measures whether SAGE can generalize beyond the specific reconstruction generator used during training.

Full CommunityForensics results. Table 5 and Table 6 report the full per-generator results on the 13 selected CommunityForensics generators. While the main paper reports representative generators for compactness, these tables provide the complete generator-wise results used to compute the average accuracy, average AUC, mAP, standard deviation, and worst-generator performance. The selected set covers diverse commercial and open-source generators, including DFGAN, DALL·E

Table 7: Cross-dataset detection performance on MS-COCOAI. For each generator, we report Acc. and AUC. Mean denotes the average performance over five diffusion-based generators.

Method	DALL·E 3		Midjourney		SD2.1		SD3		SDXL		Avg.	
	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC
B-Free	59.5	85.0	53.8	75.4	98.7	99.9	64.7	81.7	80.8	95.7	71.5	87.6
CNNSpot	49.9	36.4	50.1	63.6	50.2	62.8	50.8	66.1	50.6	70.6	50.3	59.9
FreqNet	50.1	74.8	50.0	68.6	50.0	42.4	50.2	56.0	50.2	46.1	50.1	57.6
NPR	48.8	46.5	50.7	50.0	56.7	60.6	60.8	65.8	49.2	50.4	53.2	54.7
UniFD	48.8	38.8	49.5	53.3	56.9	79.7	54.0	74.5	54.8	73.9	52.8	64.0
VIBNet	69.7	87.7	76.4	92.9	77.6	93.5	76.9	93.2	79.0	94.3	75.9	92.3
Effort	54.3	81.1	71.9	95.9	58.2	86.4	56.7	86.1	60.2	88.2	60.3	87.5
Ours	94.9	99.5	80.0	95.3	93.8	99.0	96.2	99.7	98.9	100.0	**92.7**	**98.8**

Table 8: Cross-dataset detection performance on WildRF. For each platform, we report Acc. and AUC. Mean denotes the average performance over three social media platforms. The best results are highlighted in bold.

Method	Facebook		Reddit		Twitter		Avg.	
	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC
B-Free	78.3	92.5	90.6	96.8	84.1	96.6	**84.3**	**95.3**
CNNSpot	50.0	38.1	51.1	46.0	35.2	49.9	45.4	44.7
FreqNet	59.7	78.6	60.4	65.5	39.7	50.8	53.3	65.0
NPR	64.1	78.1	53.7	63.2	62.9	53.8	60.2	65.0
UniFD	49.1	39.5	60.1	65.1	45.8	57.7	51.7	54.1
VIBNet	66.6	74.4	75.3	81.9	63.9	73.7	68.6	76.7
Effort	60.6	62.4	72.7	81.4	54.9	63.9	62.7	69.2
Ours	67.8	75.8	68.7	76.7	72.0	78.4	69.5	77.0

3, DeciDiffusionV2, FLUX-dev, FLUX-schnell, Firefly, Hourglass, IdeogramV1, Imagen3, LCM-LoRA-SDv1.5, MidjourneyV5.2, Kandinsky-2.2, and Stable Cascade.

Across the 13 generators, SAGE achieves the highest average accuracy of 84.9%, the highest average AUC of 92.3%, and an mAP of 90.3%. More importantly, SAGE shows the lowest generator-wise standard deviation in accuracy, 5.1%, and the highest worst-generator accuracy, 71.7%. These results indicate that SAGE is not optimized for only a small subset of generators, but instead provides more balanced performance across diverse unseen generator sources. Although some baselines achieve very high accuracy on specific generators, their performance varies substantially across sources. For example, B-Free performs strongly on several generators but drops on Imagen3 and IdeogramV1, while Effort achieves high AUC on many diffusion-based generators but exhibits lower accuracy on Kandinsky-2.2 and Stable Cascade. In contrast, SAGE maintains consistently high accuracy and AUC across both commercial and open-source generators, supporting the cross-generator stability of SAGE.

MS-COCOAI. Table 7 reports the cross-dataset performance on MS-COCOAI [30]. MS-COCOAI is a diffusion-based generator benchmark that includes DALL·E 3, Midjourney, SD2.1, SD3, and SDXL, along with COCO-sourced real images. This setting evaluates whether a detector trained with limited generation sources can generalize to diverse diffusion architectures under the same real-image distribution.

SAGE achieves the highest average accuracy and average AUC on MS-COCOAI, reaching 92.7% and 98.8%, respectively. In particular, SAGE obtains over 99% AUC on SD2.1, SD3, and SDXL, demonstrating strong generalization within the Stable Diffusion family. Unlike CNN-based or frequency-based detectors, which often remain close to chance-level accuracy, SAGE maintains stable performance across modern diffusion generators. Midjourney appears relatively more challenging in terms of accuracy, likely due to its proprietary generation and post-processing pipeline, but SAGE still achieves a high AUC of 95.3%.

WildRF. Table 8 reports the results on WildRF [2]. WildRF is an in-the-wild benchmark collected from social media platforms, providing a challenging evaluation setting because images can be affected by unknown compression, resizing, re-encoding, and platform-specific post-processing. In this setting, B-Free achieves the best overall performance, indicating that explicitly bias-free training can be particularly effective under strong real-world post-processing shifts. SAGE achieves second-tier performance with 69.5% average accuracy and 77.0% average AUC, slightly outperforming VIBNet on average.

Compared with its stronger performance on CommunityForensics and MS-COCOAI, the performance gap on WildRF suggests that SAGE is still somewhat vulnerable to uncontrolled post-processing artifacts and platform-specific distribution shifts. Nevertheless, SAGE maintains relatively balanced accuracy across Facebook, Reddit, and Twitter, and achieves the highest average accuracy and AUC among the evaluated methods except B-Free. These results indicate that SAGE remains competitive under real-world distribution shifts, but also suggest that additional augmentation or post-processing-aware training may be needed for more reliable deployment in highly compressed in-the-wild environments.

E.2 Semantic Anchor Ablation

Table 9 analyzes the effect of the decomposition direction used in the anchor path. We compare three variants: a random vector, the mean of caption embeddings from the training dataset, and the learned semantic anchor. The random vector shows low performance across all datasets, indicating that arbitrary feature subtraction is insufficient for constructing a useful authentication-oriented representation. The caption mean improves performance, suggesting that using a text-space semantic direction is beneficial. However, the learned semantic anchor consistently outperforms the fixed caption mean across CommunityForensics, MS-COCOAI, and WildRF. This confirms the importance of learning the semantic anchor to approximate paired text decomposition at inference time.

Table 9: Semantic anchor ablation. We compare different decomposition directions for caption-free inference.

Method	CommunityF.		MS-COCOAI		WildRF	
	Acc.	AUC	Acc.	AUC	Acc.	AUC
Random Vector	57.2	64.9	58.2	70.1	60.0	54.2
Caption Mean	73.9	89.5	80.5	91.7	64.4	73.6
Semantic Anchor	**84.9**	**92.3**	**92.7**	**98.8**	**69.5**	**77.0**

E.3 Additional Representation Analysis

Stage-wise feature analysis. Figure 7 presents a stage-wise analysis of SAGE features. We compare the raw CLIP image feature $\mathbf{v}_i$, the anchor residual feature $\mathbf{r}_i^q = \mathbf{v}_i - \mathbf{q}$, and the projected embedding $\mathbf{z}_i^q = f_\phi(\mathbf{r}_i^q)$. For each stage, we measure three complementary properties: semantic dependence, authentication utility, and real/fake separability.

At the projected feature stage, semantic dependence is substantially reduced. CKA decreases from 0.759 to 0.018, and dCor decreases from 0.934 to 0.211. At the same time, authentication utility

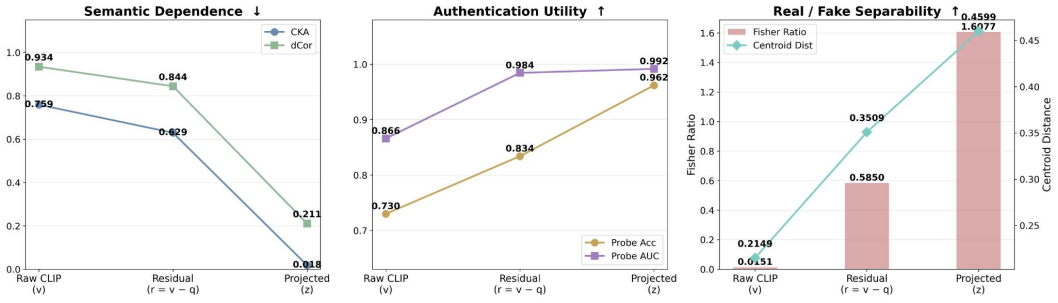

Figure 7: Stage-wise feature analysis of SAGE. As the representation progresses from Raw CLIP features to anchor residuals and projected embeddings, semantic dependence decreases, while authentication utility and real/fake separability increase.

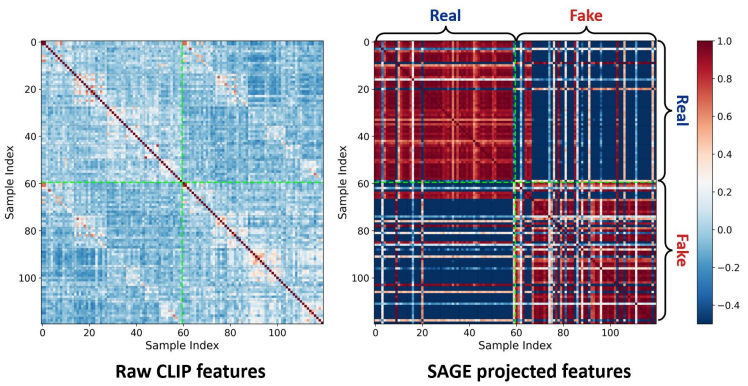

Figure 8: Centered feature cosine similarity heatmap. Samples are first sorted by real/fake labels and then by semantic clusters within each class. The green dashed lines indicate the real/fake boundary. Red denotes positive centered cosine similarity, while blue denotes negative similarity. Compared with Raw CLIP features, SAGE projected features show a clearer class-aligned block structure, indicating an authentication-oriented feature geometry.

improves: linear probe accuracy increases from 0.730 to 0.962, and probe AUC increases from 0.866 to 0.992. Separability also improves substantially, with the Fisher ratio increasing from 0.0151 to 1.6077 and the centroid distance increasing from 0.2149 to 0.4599. These results show that SAGE progressively transforms CLIP image features into representations that are less governed by text semantics and more aligned with real/fake authentication.

Feature similarity heatmap. Figure 8 visualizes how the pairwise similarity structure changes from raw CLIP features to SAGE projected features. To construct the heatmap, we first sample real and fake images from the MS-COCOAI test set and compute feature representations for each image. The samples are sorted by the real/fake label, and within each class, they are further ordered by semantic clusters obtained from CLIP text embeddings [27]. Therefore, if a feature space is mainly organized by authentication labels, the heatmap should exhibit a clear block structure aligned with the real/fake boundary. The green dashed lines indicate this real/fake class boundary.

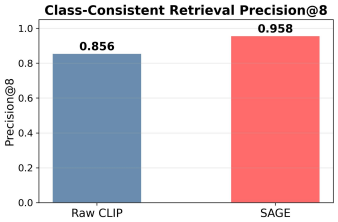

Figure 9: Class-consistent nearest-neighbor retrieval Precision@8 on the MS-COCOAI test set.

Each entry in the heatmap represents the centered cosine similarity between two samples. Red indicates positive similarity, meaning that two samples are close in the centered feature space, whereas blue indicates negative similarity, meaning that two samples are separated in opposite directions after suppressing the global mean direction. In the raw CLIP feature space, the class-aligned block structure is weak: real–real, fake–fake, and real–fake regions are not clearly separated. This suggests that raw CLIP features are not primarily organized according to real/fake authentication labels, but are still strongly affected by semantic similarity. In contrast, SAGE projected features show a much clearer block structure: within-class regions exhibit stronger positive similarity, while cross-class regions show lower or negative similarity. This indicates that SAGE reorganizes the representation space so that samples with the same authentication label become closer and samples with different labels become more separated. These results support our claim that SAGE shifts the feature geometry from semantic-dominant similarity toward authentication-oriented structure.

Class-consistent nearest-neighbor retrieval. To further evaluate the neighborhood structure of the learned feature space, we compute class-consistent Precision@8 on the full MS-COCOAI test set. A retrieved neighbor is considered class-consistent when it has the same real/fake label as the query. As shown in Figure 9, Raw CLIP achieves a Precision@8 of 0.856, whereas SAGE achieves 0.958.

Table 10: Independent semantic probing and path comparison on COCO real/fake images excluded from SAGE training. Semantic mAP evaluates object-category prediction, Auth. AUC evaluates real/fake discrimination, and CKA compares each representation with the paired CLIP text features. Metrics are reported on a 0–1 scale.

Representation	Semantic mAP ↓	Auth. AUC ↑	CKA ↓
Raw CLIP	0.769	0.866	0.759
Paired residual	0.705	0.931	0.618
Anchor residual	0.680	0.984	0.629
Paired projected feature	0.538	0.989	0.108
BLIP projected feature	0.663	0.934	0.371
SAGE (anchor path)	**0.459**	**0.992**	**0.018**

This indicates that nearest neighbors in the SAGE feature space are more likely to share the same authentication class, even when their semantic content differs.

Figure 10 and Figure 11 provide qualitative nearest-neighbor retrieval examples. Raw CLIP tends to retrieve samples that are visually or semantically similar, but the retrieved neighbors often contain mixed real/fake labels. In contrast, SAGE retrieves class-consistent neighbors across diverse semantics. This qualitative result complements the Precision@8 analysis and further shows that SAGE shifts the neighborhood structure from semantic similarity toward forensic consistency.

E.4 Independent Semantic Probing and Nonlinear Dependence

Independent semantic probing and path comparison. We evaluate semantic accessibility using independent multi-label probes trained on frozen representations to predict human-annotated COCO object categories [18]. The probing images are excluded from SAGE training. We compare Raw CLIP features, paired and anchor residuals, and projected features obtained using ground-truth captions, BLIP-generated captions [16], or the learned anchor. For each representation, Table 10 reports semantic mAP, authentication AUC, and CKA with the paired CLIP text features.

Relative to Raw CLIP, SAGE reduces semantic mAP from 0.769 to 0.459 while increasing authentication AUC from 0.866 to 0.992. Authentication AUC is already high at the anchor-residual stage (0.984), whereas projection further reduces semantic mAP from 0.680 to 0.459 and CKA from 0.629 to 0.018. The anchor and ground-truth-caption paths yield numerically similar authentication AUCs of 0.992 and 0.989, respectively, while the anchor path exhibits lower semantic mAP and CKA. The BLIP-caption alternative achieves an authentication AUC of 0.934. These results show that the caption-free anchor path retains high authentication utility while reducing semantic accessibility under the evaluated probes.

Table 11: RBF-kernel HSIC evaluated on the same held-out samples. Gaussian-kernel bandwidths are selected using the median heuristic.

Representation	RBF-kernel HSIC ↓
Raw CLIP	0.678
SAGE	**0.023**

Nonlinear dependence evaluation. To complement the linear-HSIC training objective, we evaluate nonlinear dependence using RBF-kernel HSIC [8]. Raw CLIP and SAGE are evaluated on the same held-out samples using Gaussian kernels, with bandwidths selected by the median heuristic. As shown in Table 11, RBF-kernel HSIC decreases from 0.678 for Raw CLIP to 0.023 for SAGE. This result complements the dCor reductions observed in the main analysis and the stage-wise analysis in Appendix E.3. Together, these measurements provide empirical evidence of reduced dependence beyond linear cross-covariance, but do not establish complete semantic independence.

E.5 Additional Evaluations under Distribution Shifts

Content-category and generator shifts. We restrict training to two COCO content categories and evaluate on categories disjoint from those used for training. The training fake images are SDXL reconstructions, whereas the evaluation fake images are generated by FLUX-dev through text-to-image generation using prompts from the held-out categories. Evaluation real images are drawn

from the corresponding held-out COCO categories. This setting therefore introduces simultaneous content-category and generator shifts.

As shown in Table 12, SAGE achieves 92.7% accuracy and 95.4% AUC, compared with 49.8% accuracy and 58.1% AUC for Raw CLIP. These results support improved transfer under the combined shift. However, the experiment does not isolate the effects of content, generator, style, appearance, or image quality, and therefore does not rule out all semantic shortcuts.

Table 12: Detection under simultaneous content-category and generator shifts. Training uses two COCO categories with SDXL reconstructions; evaluation uses disjoint COCO categories and FLUX-dev text-to-image outputs. Accuracy and AUC are reported in percent.

Method	Avg. Acc. ↑	Avg. AUC ↑
Raw CLIP	49.8	58.1
SAGE	**92.7**	**95.4**

E.6 Robustness to Post-Processing

We evaluate robustness to JPEG compression with a quality factor of 50, Gaussian blur, 0.5× downscaling, and Gaussian noise. We compare SAGE with UniFD [21], B-Free [9], and Effort [40] to examine their behavior under these post-processing operations.

As shown in Table 13, SAGE achieves AUCs of 93.0%, 91.6%, 90.0%, and 87.5% under JPEG compression, blur, downscaling, and noise, respectively, giving the highest AUC among the compared methods in each condition. Its accuracy remains strong or competitive across the evaluated operations. These results support robustness under the tested perturbations, but do not cover arbitrary transformation strengths or unknown combinations of post-processing operations. SAGE also remains below B-Free on WildRF (Table 8), suggesting that additional post-processing-aware training may be beneficial under uncontrolled real-world shifts.

E.7 Training-Data Ablations

Multi-source training. We examine the effect of additional fake-image sources by retraining SAGE with SDXL [24], DDPM [11], and GauGAN data [23]. Table 14 compares the original SDXL-only model with the 45K and 90K multi-source settings.

In the 45K setting, average accuracy and AUC are 83.9% and 92.0%, compared with 84.9% and 92.3% for SDXL-only training. DFGAN accuracy increases from 87.1% to 88.2%, but the overall average and Hourglass performance do not improve. In the 90K setting, average accuracy and AUC increase to 86.4% and 95.2%, with DFGAN reaching 96.1% accuracy and 99.0% AUC and Hourglass reaching 74.0% accuracy and 78.0% AUC. These results suggest that multi-source training can benefit SAGE with greater per-source coverage. However, the 90K setting changes both source diversity and data scale, so their individual effects cannot be separated by this comparison.

Effect of semantic pairing. We examine the role of semantically aligned training data by comparing the original paired setting with an unpaired setting in which the real and fake images are not semantically aligned and the fake image is not a reconstruction of the corresponding COCO image. Both settings are evaluated on MS-COCOAI [30].

As shown in Table 15, paired training achieves 92.7% average accuracy and 98.8% average AUC, compared with 85.1% and 93.5% for unpaired training. CKA with the paired text features decreases

Table 13: Robustness to post-processing. Accuracy and AUC are reported in percent. Bold values indicate the highest accuracy or AUC among the compared methods within each evaluation condition.

	B-Free		UniFD		Effort		SAGE	
Perturbation	Acc.	AUC	Acc.	AUC	Acc.	AUC	Acc.	AUC
Clean	80.9	85.8	56.3	57.7	74.9	90.1	**84.9**	**92.3**
JPEG, QF = 50	74.9	77.8	53.4	50.0	54.3	59.6	**85.7**	**93.0**
Gaussian blur	79.0	81.4	54.7	55.6	64.7	87.4	**84.3**	**91.6**
0.5× downscaling	78.2	82.2	55.0	50.3	65.1	89.7	**82.7**	**90.0**
Gaussian noise	**78.6**	82.7	53.1	53.8	73.2	84.7	78.5	**87.5**
Avg. Δ	−3.2	−4.8	−2.2	−5.3	−10.6	−9.8	−2.1	−1.8

Table 14: Multi-source training results. Accuracy and AUC are reported in percent. The 45K and 90K labels identify the two evaluated data-budget settings.

	Average		DFGAN		Hourglass	
Fake sources	Acc.	AUC	Acc.	AUC	Acc.	AUC
SDXL only	84.9	92.3	87.1	98.2	71.7	73.1
SDXL + DDPM + GauGAN (45K)	83.9	92.0	88.2	98.2	68.3	70.3
SDXL + DDPM + GauGAN (90K)	**86.4**	**95.2**	**96.1**	**99.0**	**74.0**	**78.0**

Table 15: Effect of semantically paired training data on MS-COCOAI. Accuracy and AUC are reported in percent; CKA is reported on a 0–1 scale.

Training data	Avg. Acc. ↑	Avg. AUC ↑	CKA ↓
Unpaired	85.1	93.5	0.231
Paired	**92.7**	**98.8**	**0.051**

from 0.231 in the unpaired setting to 0.051 in the paired setting. These results support the benefit of semantically aligned data construction for detection performance and reduced measured alignment with caption features under the evaluated conditions.

E.8 Additional Architectural Ablations

Comparison of residualization operators. Using the same projection module, we compare caption-feature subtraction with an orthogonal projection that removes the component of the image feature aligned with the normalized caption direction. As shown in Table 16, orthogonal projection achieves 83.7% average accuracy and 91.9% average AUC, while subtraction achieves 84.9% and 92.3%, respectively. Both variants outperform the Raw CLIP baseline, which achieves 73.7% accuracy and 85.5% AUC. The two operators yield broadly similar detection performance in this comparison. We retain subtraction because of its simplicity and its higher observed performance in the evaluated setting.

Number of sub-centers on CommunityForensics. We repeat the sub-center analysis on CommunityForensics, comparing $K = 1, 2, 4$, and 6 with a BCE classifier without proxies. K denotes the number of sub-centers per real/fake class rather than the number of generators.

As shown in Table 17, $K = 2$ achieves the highest average accuracy and AUC at 84.9% and 92.3%. Increasing K to 4 or 6 reduces performance to 82.1% / 91.1% and 79.7% / 88.4% accuracy / AUC, respectively. This trend is consistent with the MS-COCOAI analysis in Figure 6 and supports the choice of $K = 2$ for the evaluated benchmarks. It also indicates that increasing the number of class centers does not necessarily improve cross-generator performance.

F Limitations and Societal Impact

Limitations. SAGE relies on a frozen CLIP image–text backbone and may inherit its representation biases and failure modes. Moreover, our decomposition is motivated by the aligned CLIP image–text space, but we do not assume that image and text embeddings lie in an identical semantic subspace.

Table 16: Comparison of feature-processing variants. Subtraction and orthogonal projection use the same projection module. Accuracy and AUC are reported in percent.

Decomposition method	Avg. Acc. ↑	Avg. AUC ↑
Raw CLIP	73.7	85.5
Projection only	78.8	87.1
Orthogonal projection	83.7	91.9
Subtraction	**84.9**	**92.3**

Table 17: Effect of the number of sub-centers on CommunityForensics. K denotes the number of sub-centers per real/fake class. Accuracy and AUC are reported in percent.

Classifier setting	Avg. Acc. ↑	Avg. AUC ↑
BCE, no proxy	81.5	91.5
$K = 1$	81.9	91.5
$K = 2$	**84.9**	**92.3**
$K = 4$	82.1	91.1
$K = 6$	79.7	88.4

Prior work has shown that multimodal contrastive representations can exhibit a modality gap [17]. Therefore, the image–text subtraction used in SAGE should be interpreted as a semantically guided residualization step rather than a complete removal of semantic information. Although HSIC-based semantic decorrelation further reduces statistical dependence between image-side features and text embeddings, it does not guarantee complete removal of all task-irrelevant semantic information.

In addition, our paired training data are constructed from COCO images and SDXL reconstructions, which do not cover all possible image domains, editing operations, post-processing pipelines, or future generator artifacts. Finally, the semantic anchor is a single learnable surrogate vector, and more adaptive test-time decomposition mechanisms may further improve robustness under severe domain shifts.

Societal impact. AI-generated image detection can have positive societal impacts. Reliable detectors can help mitigate risks such as misinformation, identity forgery, copyright infringement, and manipulation of digital evidence. In real-world scenarios, the source generator of an image is often unknown, making it important for detectors to operate reliably across diverse unseen generators. Since SAGE is designed to reduce reliance on semantic shortcuts and lower generator-wise performance variation, it can contribute to building more stable media forensics systems.

However, such technology should be used with caution. AI-generated image detectors are not perfect and can produce both false positives and false negatives. False positives may incorrectly classify real images as synthetic, potentially harming creators, journalists, or ordinary users. Conversely, false negatives may allow harmful synthetic content to evade detection. Therefore, predictions from SAGE should be used as decision-support signals rather than as the sole basis for high-stakes decisions, and should be considered together with provenance metadata, human review, and contextual evidence.

Detection technology also involves dual-use risks. Stronger detectors can help reduce the misuse of synthetic content, but they may also provide adversaries with insights into the features that detectors rely on and their failure modes, enabling the development of stronger evasion strategies. Therefore, when deploying or evaluating AI-generated image detectors, it is important to transparently report not only performance numbers but also failure cases, dataset coverage, confidence calibration, and the intended scope of use. We believe that research on generalized AI-generated image detection can be socially beneficial, but its impact becomes more trustworthy when model limitations are clearly communicated and appropriate deployment safeguards are considered.

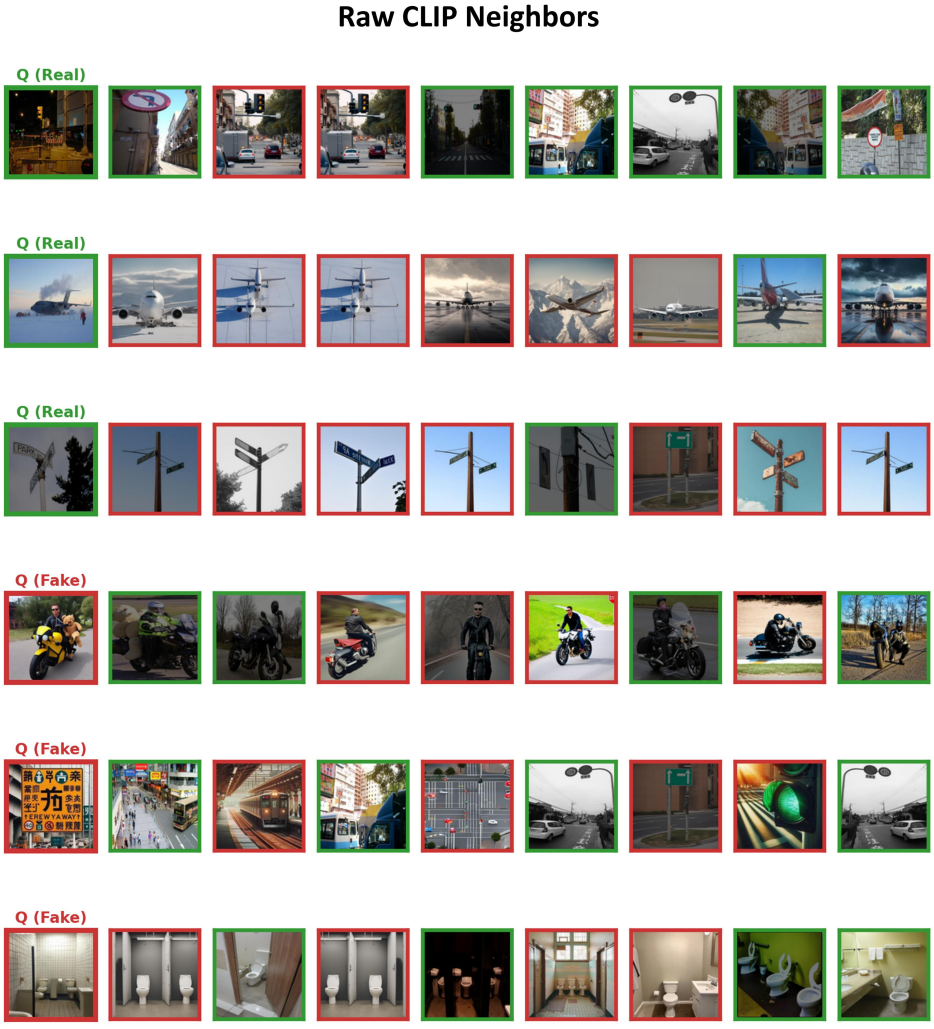

Figure 10: Nearest-neighbor retrieval examples in the Raw CLIP feature space. For each query image, we show the top-8 retrieved neighbors based on cosine similarity. Green and red borders indicate real and fake images, respectively. Raw CLIP tends to retrieve visually or semantically similar samples, such as images with similar objects, scenes, or layouts, but the retrieved neighbors often contain mixed real/fake labels. This shows that the neighborhood structure of Raw CLIP is strongly influenced by semantic similarity rather than authentication consistency.

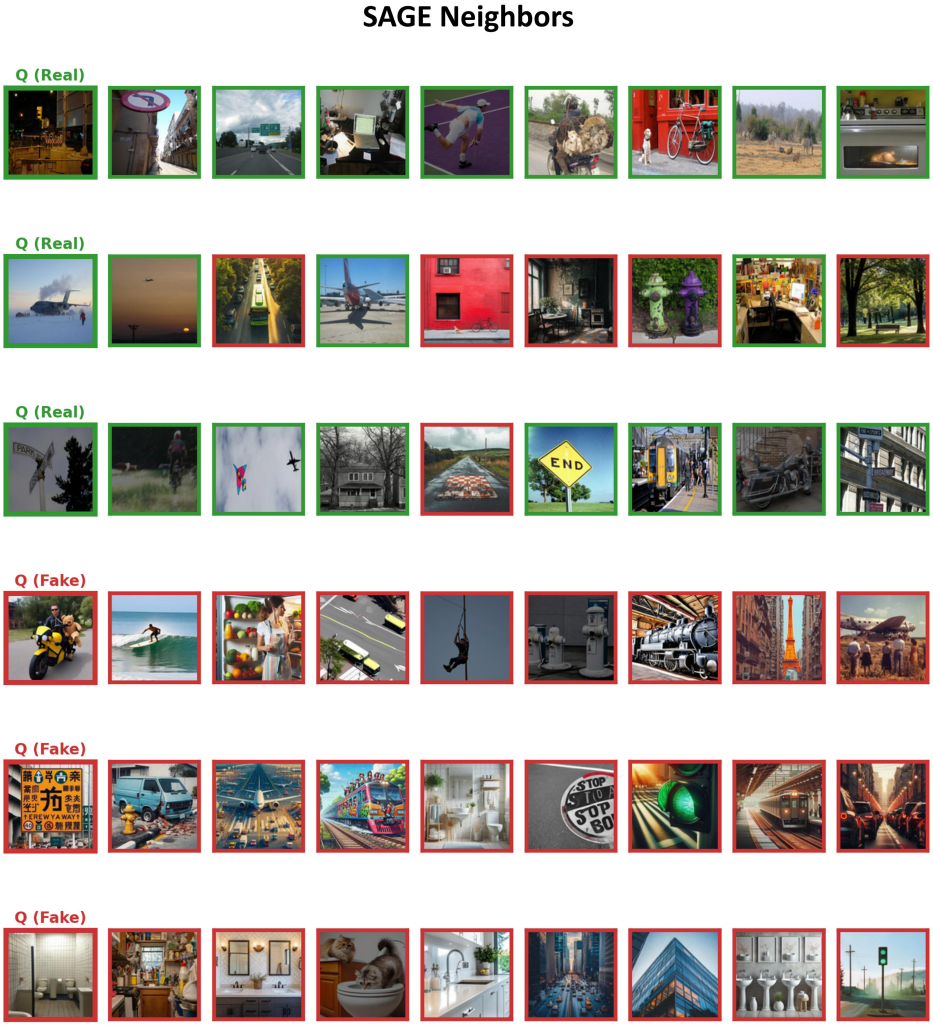

Figure 11: Nearest-neighbor retrieval examples in the SAGE feature space. For the same query images as in Figure 10, we show the top-8 retrieved neighbors using SAGE projected embeddings. Green and red borders indicate real and fake images, respectively. Unlike Raw CLIP, SAGE retrieves neighbors that are more consistent with the query's authentication label, even when their semantic content differs substantially. This indicates that SAGE reorganizes the feature space from semantic similarity toward forensic consistency and authentication-oriented neighborhood structure.

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: The claims are stated in the abstract and introduction and are supported by the main and appendix experiments.

   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: Limitations are discussed in the main text and further detailed in Appendix F, covering semantic dependence, data coverage, and generalization challenges.

   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: Appendix C provides the assumptions and proof for the linear HSIC and cross-covariance relation.

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: The method, data construction, metrics, and implementation details needed for reproduction are described in the main paper and Appendix D.

   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: The SAGE implementation is publicly available on GitHub, and the paper provides experimental details to support reproducibility.

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: Training data, benchmarks, metrics, hyperparameters, optimization, and inference details are specified in the paper and Appendix D.

   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: We report generator-wise distributions, standard deviation, and worst-generator performance to measure cross-generator variability.

   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: Appendix D reports the GPU type, training time, optimizer, batch size, and training epochs.

   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: The work follows the NeurIPS Code of Ethics and discusses relevant risks in Appendix F.

   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: Broader impacts are discussed in the main text and further detailed in Appendix F, covering positive applications, detection errors, and dual-use risks.

    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: The paper does not release a high-risk generator or scraped dataset; dual-use risks are discussed in Appendix F.

    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: Existing datasets and models are cited, and Appendix D summarizes their usage terms.

    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 public dataset, benchmark, codebase, or model checkpoint is introduced as a primary submitted 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: The paper does not involve crowdsourcing or human-subject studies.

    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: The paper does not involve human-subject studies.

    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: [N/A]

    Justification: LLMs are not used as core method or experiment components; any use is limited to writing or formatting assistance.

    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.