SimLBR: Learning to Detect Fake Images by Learning to Detect Real Images

Supplementary Material

	GenImage		AIGC	
	ID	OD	ID	OD
w/o LBR	94.37	77.70	85.58	73.26
w/ LBR	95.47	88.00	91.50	85.98
Gain	1.10	**10.30**	5.92	**12.72**

Table 6. **Average accuracy on ID vs. OD testsets:** We ablate the impact of using Latent Blending Regularization on In-Distribution (ID) and Out-Distribution (OD) generative testsets. ID testset contains generators that share architectural similarities with the training generator, whereas OD testset contains generators with substantially different architectures. We observe the most significant gains from LBR in the OD testset, reasserting that LBR encourages the model to learn a generator-agnostic decision boundary.

7. Additional Implementation Details

To simplify implementation, instead of resampling different labels for the same real image during training, we iterate over the entire dataset and, whenever a fake image is encountered, we sample a real image to form a pair and assign it a fake label. This pairing strategy also ensures that all real and fake samples in the dataset are used during training.

8. Evaluating LBR on Out of Distribution Generators

In Table 5, we presented the gains in average accuracy obtained by adding Latent Blending Regularization (LBR). Prior works [49, 57] have shown that detectors generally perform better on generators that share architectural similarities with the training generator, as they can share similar artifacts in their generated data. Thus, to further isolate the contribution of LBR, we partition the test-time generators into two groups: In-Distribution (ID) and Out-Distribution (OD).

The ID set contains generators whose architectural families match that of the training generator, while the OD set contains generators with fundamentally different architectures. For the AIGC benchmark, all GAN-based generators are treated as ID and diffusion-based generators as OD; for GenImage, this assignment is reversed. This partition allows us to assess whether LBR improves performance primarily on generators that are artifact-similar (ID) or on generators that introduce entirely different characteristics (OD).

Table 6 shows the average accuracy across different settings. In both GenImage and AIGC benchmarks, LBR yields modest gains on the ID set, where the baseline already performs well. However, the improvements on the OD set are substantially larger, with gains of 10% on GenImage and 12.72% on AIGC. These results indicate that while baseline detectors overfit to artifacts present in the training generator, LBR helps the model learn a more principled separation between the real and fake distributions. As a result, SimLBR achieves significantly stronger performance on OD generators, demonstrating its ability to generalize beyond artifact-level correlations and to adapt to previously unseen generative architectures.

9. Robustness to Image Perturbations

Prior works [49, 54] show that detectors exhibit severe performance degradation when perturbations such as JPEG Compression and Gaussian Blur are applied to the input images. We evaluate the performance of SimLBR under different compression and blur settings on the AIDE benchmark. The reported accuracy is averaged across 16 different generative models. The results are summarized in Table 7. Compared to the state-of-the-art models like AIDE [49] and PatchCraft [54], SimLBR shows extreme robustness to these perturbations. SimLBR achieves the highest accuracy across all perturbations, exhibiting incredible robustness to adversarial inputs. Furthermore, unlike most prior methods [47, 49, 54], we did not include JPEG compression or Gaussian blur augmentations during training, which indicates that SimLBR is truly robust to unseen perturbations.

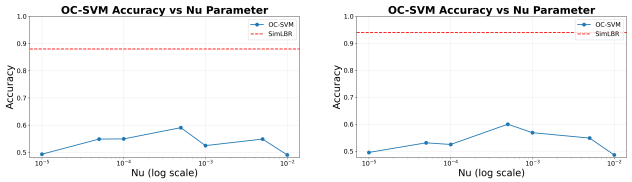
10. Comparison to Anomaly Detection

A natural strategy to avoid overfitting to generator–specific artifacts is to model only the real image distribution and treat AI-generated image detection as an anomaly-detection problem. To evaluate this idea, we train a One-Class SVM (OC-SVM) using DINOv3 features of real training images for both the AIGC and GenImage benchmarks. We use the *RBF* kernel and do a sweep to find the best ν. As shown in Figure 4, the OC-SVM performs substantially worse than SimLBR across both settings.

This result highlights a fundamental limitation of pure anomaly-detection approaches in high-dimensional image spaces. Modern generative models produce highly photorealistic images whose latent representations often lie close to those of real images. As a result, simply learning the support of the real distribution is insufficient; fake images do not reliably fall outside this region. In contrast, SimLBR introduces structured guidance by injecting controlled amounts of fake latent information into real samples during training. This guided perturbation shapes a more meaningful decision boundary around the true real-image

Method	Original	JPEG Compression				Gaussian Blur			
		QF=95	QF=90	QF=75	QF=50	$\sigma = 1.0$	$\sigma = 2.0$	$\sigma = 3.0$	$\sigma = 4.0$
CNNSpot	70.78	64.03	62.26	60.65	59.66	68.39	67.26	67.13	65.85
FreDect	64.03	66.95	67.45	66.64	65.33	65.75	66.48	68.58	69.64
Fusing	68.38	62.43	61.39	59.34	57.41	68.09	66.69	66.02	65.58
LNP	83.84	53.58	54.09	53.02	52.85	67.91	66.42	66.2	62.69
LGrad	75.34	51.55	51.39	50.00	50.00	71.73	69.12	68.43	66.22
DIRE-G	68.68	66.49	66.12	65.28	64.34	64.00	63.09	62.21	61.91
UnivFD	78.43	74.10	74.02	69.92	68.68	70.31	68.29	64.62	61.18
PatchCraft	89.31	72.48	71.41	69.43	67.78	75.99	74.90	73.53	72.28
AIDE	**92.77**	75.54	74.21	70.64	69.60	81.88	80.35	80.05	79.86
SimLBR	88.40	**86.04**	**83.92**	**78.98**	**73.48**	**88.03**	**86.65**	**86.39**	**86.30**

Table 7. **Robustness to JPEG Compression and Gaussian Blur:** Following prior work, we evaluate the robustness of our model when different amounts of JPEG Compression and Gaussian Blur are applied to the input images. SimLBR is highly robust, showing minimal degradation even under strong image perturbations.



(a) AIGC (b) GenImage

Figure 4. **Comparison with OC-SVM:** We train a One-Class SVM using ProGAN and SD v1.4 images and evaluate on AIGC and GenImage benchmarks, respectively. SimDLR clearly outperforms the outlier detector approach when using the same latent features.

manifold, enabling the detector to distinguish real from fake images even when fakes closely mimic real data.

11. Visualization of SimLBR Embeddings

Figure 6 shows t-SNE visualizations of embeddings from final deep layer of the MLP trained with and without Latent Blending Regularization. For the training set, we use images generated by the ProGAN model and visualize embeddings across five datasets with diverse generated images. We see that naively training the model to differentiate between real and fake images results in models that overfit to the training data and do not preserve meaningful structures between real and fake distributions. LBR forces the model to learn a more meaningful separation between the real and generated image distribution by learning a tight boundary around the real images, which is visible as the compact blue/orange cluster. These visualizations strongly suggest that LBR appropriately regularizes the training objective, forcing the model to learn generator-agnostic decision boundaries.

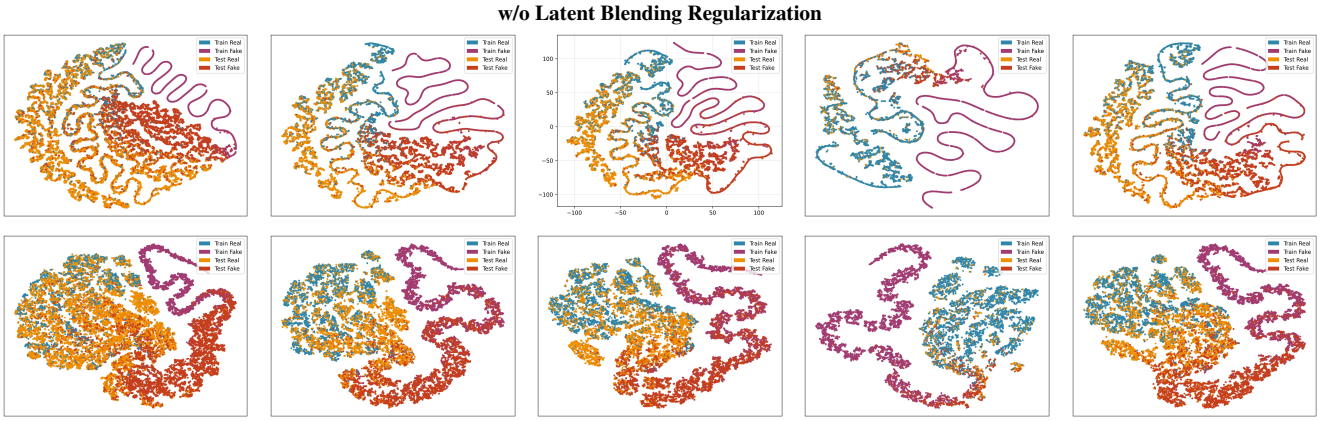
12. Lack of Robustness to Different Backbones

As illustrated in Table 5, while LBR yields significant performance gains when utilizing the frozen DINOv3 feature space, these improvements are less pronounced when employing DINOv2. Because SimLBR operates entirely within a frozen latent space, its efficacy is inherently upper-bounded by the expressive power and geometry of the underlying manifold. While LBR provides a robust regularization framework for learning decision boundaries, it remains contingent on the quality of the pre-trained representations. However, as foundation models continue to evolve and provide increasingly discriminative latent spaces, we anticipate that the effectiveness of LBR will scale accordingly with future architectural advancements.

13. Experimental Benchmarks and Dataset Protocols

For the AIGC benchmark, we utilize real images from the LSUN dataset and synthetic counterparts generated via ProGAN for training. To rigorously evaluate generalization, the test set incorporates real images from five diverse sources, ImageNet, COCO, LSUN, FFHQ, and CelebA, while the fake images are sourced from 16 distinct generative models.

In the GenImage setup, real images for both training and testing are derived from ImageNet. The synthetic training data is generated using Stable Diffusion (SD) 1.4. The evaluation suite comprises fake images from 8 different generative architectures, allowing for a robust assessment of model performance against unseen generative distributions.



(a) Chameleon (b) Wukong (c) Glide (d) DALLE2 (e) ADM

Figure 6. **t-SNE plot visualization:** Figure shows t-SNE plot from the penultimate layer of models trained with and without our approach. While the existing approach overfits to fake samples in the training set, our method correctly models the separation between real and fake image distributions, allowing generalization to any generator.

14. RSFake-1M Dataset

In addition to evaluating on natural-image benchmarks such as AIGC and GenImage, we also conduct experiments on RSFake-1M [42], a large-scale dataset designed for detecting diffusion-generated satellite imagery forgeries. The objective of including this dataset is to evaluate whether LBR improves performance across highly varied image distributions. RSFake-1M contains 500,000 synthetic satellite images generated from 10 satellite-specific diffusion models, paired with 500,000 authentic satellite images. The synthetic generators include: DiffusionSat-512 [20], DiffusionSat-256 [20], GeoRSSD [53], SD-FRS [50], GeoSynth-Text [33], GeoSynth-Sam [33], GeoSynth-Canny [33], CRSDiff [44], MapSat [6], and RSPaint [15].

For our experiments, we train SimDLR using DiffusionSat-512 and DiffusionSat-256, and evaluate on held-out test sets from all ten models to assess cross-generator generalization. As the detection backbone, we use the DINOv3-Satellite model, pretrained on 493M satellite images. Since a satellite-specific version of DINOv2 is not available, the RSFake-1M evaluations are performed using only the DINOv3 backbone.