Region-Preserving Latent Counterfactual Explanations for Image Classification

A Grad-CAM-Guided Latent Optimization Approach

Rinaldi Pramandika Komber[1*], I Wayan Ivan Zenatmaja[1], Abdurrahman Harish AlMauqy[1], Wayan Firdaus Mahmudy[1], and Novanto Yudistira[1]

Faculty of Computer Science, Brawijaya University, Malang, Indonesia
rinaldikomber@student.ub.ac.id, iwayanivanz@student.ub.ac.id, abdurrahmanh@student.ub.ac.id, wayanfm@ub.ac.id, yudistira@ub.ac.id

* Corresponding author

Abstract. Counterfactual explanations help users understand image classifiers by showing how an input would need to change to alter a model prediction. Pixel-space adversarial methods achieve high flip rates, but their perturbations are often imperceptible and semantically meaningless. Latent-space generative approaches produce more natural images, yet without spatial constraints they may modify background regions unrelated to the object of interest. We propose a region-preserving latent counterfactual explanation method that optimizes a VAE latent code while keeping the classifier and generator frozen, using a Grad-CAM-derived mask to penalize changes outside salient object regions. The objective combines target-class flip loss, pixel similarity, latent proximity, perceptual similarity, background preservation, and total variation regularization. Experiments on STL-10 with an EfficientNet-B0 classifier demonstrate that the proposed method preserves the high flip rate of latent counterfactuals while improving spatial localization and background stability relative to unconstrained latent perturbation. Ablation studies quantify the roles of perceptual and background-preservation terms.

Keywords: Counterfactual explanation · Explainable AI · Latent space optimization · Variational autoencoder · Region preservation · Grad-CAM

1 Introduction

Deep neural networks have become standard tools for image classification, but their predictions remain difficult to interpret [20]. Counterfactual explanation is an intuitive approach: instead of only highlighting discriminative regions, it asks what minimal visual change would alter the classifier's decision, presenting a modified image that flips the prediction while preserving as much original content as possible [1].

© The Author(s) 2026
M. Murtadha et al. (eds.), *Proceedings of the 2026 11th International Conference on Sustainable Information Engineering and Technology (SIET 2026)*, Atlantis Highlights in Computer Sciences 26,
https://doi.org/10.2991/978-94-6239-828-3_3

A common way to produce counterfactual images is to perturb the input directly in pixel space. Methods derived from adversarial attacks, such as FGSM [6] and PGD [7], can change model predictions efficiently. However, the resulting perturbations often appear as high-frequency noise or imperceptible changes that do not correspond to meaningful semantic changes. Generative counterfactuals address this by optimizing in the latent space of a learned generator, such as a variational autoencoder [3]. These methods tend to produce more natural images, but they do not guarantee that changes occur in object-relevant regions. Background and irrelevant textures may change, which weakens the explanatory value.

Recent advancements in explainable artificial intelligence (XAI) have shifted toward generative approaches to produce counterfactual explanations (CFEs) that are both realistic and actionable [13,14,15]. By leveraging Variational Autoencoders (VAEs) and other deep generative models, researchers have moved beyond pixel-space perturbations—which often produce noisy or imperceptible changes—to optimize within structured latent spaces, ensuring that generated counterfactuals remain on the data manifold. Notable works by Verma et al. [1] and Chowdhury et al. [13] demonstrate that traversing a learned latent distribution can efficiently identify the minimal edits required to flip a classifier's prediction while maintaining sample fidelity. Furthermore, cutting-edge techniques such as the Riemannian latent traversal proposed by Pegios et al. [14] and the integration of latent diffusion models by Nguyen et al. [8] have significantly enhanced the perceptual quality and semantic coherence of these visual explanations.

However, a persistent challenge in latent-space optimization is the lack of explicit spatial constraints, which often leads to unconstrained modifications of irrelevant image regions. While methods such as Counterfactual Latent Uncertainty Explanations (CLUE) provide robust mechanisms for exploring uncertainty within the latent space to ensure explanations are in-distribution, they do not inherently restrict where visual changes occur in the decoded output [21]. Consequently, these methods frequently alter background textures or unrelated object features, resulting in explanations that may be unfaithful to the model's actual decision logic and harder for humans to interpret.

The key gap addressed in this work is spatial preservation in latent counterfactual generation. A counterfactual explanation should ideally modify visual evidence related to the class decision while preserving background content unrelated to the object. Existing latent-space methods improve visual realism but often lack an explicit constraint preventing irrelevant background modification. To bridge this gap, our work proposes a region-preserving latent counterfactual method that utilizes Grad-CAM to derive a binary saliency mask, thereby guiding the optimization to strictly localize changes within object-relevant regions while maintaining background stability.

We propose a region-preserving latent counterfactual explanation pipeline. A Grad-CAM saliency map from the original prediction is converted into a region mask that separates salient object areas from background. Counterfactual

generation is then performed by optimizing the VAE latent vector under a multi-component loss that encourages target-class prediction while penalizing unnecessary latent, perceptual, and background changes. The contributions are: (i) a lightweight latent counterfactual pipeline suitable for resource-constrained experimentation; (ii) a Grad-CAM-guided region-preservation term that discourages background drift; and (iii) an empirical evaluation on STL-10 using flip rate, similarity, perceptual quality, background preservation, runtime, and ablation studies.

2 Related Work

Adversarial attacks can be repurposed as counterfactual explanations. FGSM applies a single-step gradient update along the sign of the loss gradient [6], while PGD extends this with iterative projections onto an L_∞ ball [7]. While these methods maximize flip rate under minimal distance, the resulting perturbations are not human-interpretable and lack semantic structure. In this work, FGSM and PGD serve as adversarial reference baselines rather than realism baselines, because their near-imperceptible noise does not produce semantically coherent target-class images.

VAEs and VQ-VAEs learn compact, structured latent representations [10]. Counterfactual generation by latent traversal or optimization ensures outputs remain realistic. Recent work by Nguyen et al. [8] combined latent diffusion with visual explanations for counterfactual synthesis, while Pegios et al. [9] explored classifier-guided diffusion for counterfactual generation. Bedel and Çukur [16] further extended diffusion-driven counterfactual explanation to functional MRI, demonstrating the generality of latent-space generative approaches across imaging modalities. However, these approaches rarely enforce spatial localization of changes, often modifying background clutter. Our work adds an explicit spatial constraint to latent optimization, penalizing background deviation while preserving the generative realism of VAE-based counterfactuals.

Gradient-weighted Class Activation Mapping (Grad-CAM) produces coarse localization maps highlighting discriminative regions [2]. Subsequent variants such as Grad++ ScoreCAM [18] and AdvancedScoreCAM [19] have further improved the fidelity and resolution of gradient-based saliency maps. We leverage Grad-CAM to define object versus background masks, allowing our optimization to explicitly penalize background deviation—a design choice that complements existing latent counterfactual methods rather than replacing them.

3 Methodology

Our system comprises three frozen components: (1) a CNN classifier f; (2) a VAE with encoder E and decoder D; and (3) a region-preserving latent optimizer. Only the latent vector $\mathbf{z}$ is updated during counterfactual generation.

3.1 CNN Classifier

We employ EfficientNet-B0 [4] fine-tuned on STL-10 and subsequently frozen. The classifier outputs logits $f(x) \in \mathbb{R}^{10}$ for the ten STL-10 classes. Training uses the Adam optimizer with learning rate 10^{-3} for 30 epochs. Only correctly classified test images are used for counterfactual evaluation to ensure that a genuine flip of prediction is measured. The classifier achieves approximately 91–92% validation accuracy.

3.2 Variational Autoencoder

A standard convolutional VAE is trained on the STL-10 training split. The encoder compresses an input image x into a latent distribution $q(\mathbf{z}|x)$; we use the mean vector $\mathbf{z}_0$ as the initial latent code. The decoder reconstructs $\hat{x} = D(\mathbf{z})$ [3]. The latent dimension is 128. Both E and D are frozen during counterfactual optimization to preserve manifold structure.

3.3 Region-Preserving Latent Optimization

Given an input x with true class c_{orig} and target class c_{target} (top-2 competing class), we seek a perturbed latent code $\mathbf{z}^* = \mathbf{z}_0 + \delta\mathbf{z}$ such that $f(D(\mathbf{z}^*))$ predicts c_{target}. To ensure changes are localized to the object region, we extract a binary region mask M from Grad-CAM applied to the original image and original class:

$$M = \text{Grad-CAM}(f, x, c_{\text{orig}}) > \tau, \tag{1}$$

where τ is the 70th percentile of the Grad-CAM activation values. Pixels where $M = 1$ denote the object region; $M = 0$ denotes background. We use the final convolutional layer of EfficientNet-B0 for Grad-CAM extraction, with ReLU normalization on the gradient-weighted feature maps.

3.4 Multi-Component Loss Function

The total loss driving latent optimization is:

$$\begin{aligned}\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{class}} + \lambda_{\text{pixel}}\mathcal{L}_{\text{pixel}} + \lambda_{\text{latent}}\mathcal{L}_{\text{latent}} \\ + \lambda_{\text{LPIPS}}\mathcal{L}_{\text{LPIPS}} + \lambda_{\text{bg}}\mathcal{L}_{\text{background}} + \lambda_{\text{tv}}\mathcal{L}_{\text{TV}}.\end{aligned} \tag{2}$$

Table 1 summarizes each component. All loss weights are fixed: $\lambda_{\text{pixel}} = 1.0$, $\lambda_{\text{latent}} = 0.5$, $\lambda_{\text{LPIPS}} = 1.0$, $\lambda_{\text{bg}} = 2.0$, $\lambda_{\text{tv}} = 0.1$.

Optimization uses Adam with learning rate 0.01 for 100 iterations. Early stopping is applied if the target class is predicted with confidence > 0.9 for 10 consecutive steps. All experiments use a fixed random seed (42) for reproducibility.

Table 1: Components of the total loss function.

Component	Formula	Purpose
$\mathcal{L}_{\text{class}}$	$\text{CE}(f(D(\mathbf{z})), c_{\text{target}})$	Encourage prediction flip
$\mathcal{L}_{\text{pixel}}$	$\|x - D(\mathbf{z})\|_1$	Pixel-level minimality
$\mathcal{L}_{\text{latent}}$	$\|\mathbf{z} - \mathbf{z}_0\|_2^2$	Latent proximity
$\mathcal{L}_{\text{LPIPS}}$	$\text{LPIPS}(x, D(\mathbf{z}))$	Perceptual quality
$\mathcal{L}_{\text{background}}$	$\|(1 - M) \odot (x - D(\mathbf{z}))\|_1$	Background preservation
$\mathcal{L}_{\text{TV}}$	$\sum_{i,j} \|\nabla D(\mathbf{z})_{i,j}\|_1$	Total variation smoothness

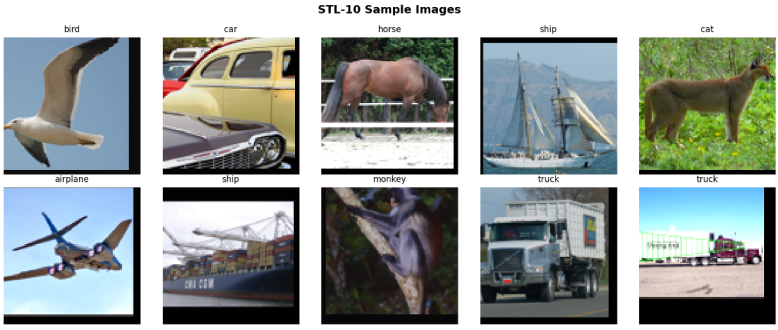


Fig. 1: Representative samples from the STL-10 dataset across all ten classes.

3.5 Baseline Methods

We compare five configurations: **M1 (FGSM)**: single-step adversarial perturbation in pixel space (adversarial reference); **M2 (PGD)**: iterative adversarial perturbation in pixel space (adversarial reference); **M3 (VAE Random)**: random perturbation in VAE latent space without directional optimization (generative baseline) [11]; **M4a (Unconstrained Latent)**: latent optimization without the Grad-CAM mask ($\lambda_{\text{bg}} = 0$, generative baseline for ablation); **M4 (Proposed)**: region-preserving latent optimization with the full loss. Adversarial methods (M1, M2) optimize for minimal pixel distance and high flip rate but do not produce semantically meaningful target-class images. Generative methods (M3, M4a, M4) trade some pixel-level similarity for semantic coherence and class-conditional realism.

4 Experimental Setup

4.1 Dataset

Experiments are conducted on STL-10, a 10-class natural image dataset (96×96 resolution) with an 80%/10%/10% train/validation/test split (seed 42). Representative samples from all ten classes are shown in Fig. 1.

4.2 Metrics

We adopt: (1) *Prediction metrics:* Flip Rate and Target Confidence. (2) *Image quality metrics:* SSIM [12] (higher is better), L2 Distance (lower is better), and LPIPS [5] (lower is more perceptually similar). (3) *Region preservation metrics:* BG-SSIM (SSIM on background pixels only, higher is better), Outside-Mask L1 (lower is better), Inside/Outside Change Ratio (higher indicates more localized changes), and Mask Leakage (percentage of background pixels that change by more than threshold $|x - \hat{x}| > 0.1$, lower is better). (4) *Efficiency metrics:* Runtime (average seconds per image on Google Colab T4 GPU). All metrics are computed on 100 correctly classified test images.

4.3 Implementation

The CNN is fine-tuned for 30 epochs with Adam (lr = 10^{-3}, batch size 32). The VAE has encoder channels [32, 64, 128, 256] and decoder channels [256, 128, 64, 32], with latent dimension 128, trained for 100 epochs with Adam (lr = 10^{-3}). For M4, we optimize $\mathbf{z}$ for 100 iterations using Adam with lr = 0.01. Loss weights: $\lambda_{\text{pixel}} = 1.0$, $\lambda_{\text{latent}} = 0.5$, $\lambda_{\text{LPIPS}} = 1.0$, $\lambda_{\text{bg}} = 2.0$, $\lambda_{\text{tv}} = 0.1$. Grad-CAM uses the final convolutional layer of EfficientNet-B0 with $\tau = 70$th percentile.

All evaluation metrics are computed using a subset of 100 test images that were correctly classified by the classifier model. The adoption of this limited sample size is a standard practice within the counterfactual explanation literature due to significant computational constraints, as the iterative optimization process required for each individual image is highly resource-intensive and time-consuming. This technical justification is supported by Antorán et al. [21] and Goyal et al. [22], who both emphasize that per-instance gradient-based optimization or latent space traversal requires substantial computation, often making evaluation across an entire large-scale dataset unfeasible. Furthermore, the selection of 100 samples is statistically defensible; based on Cochran's formula, a sample size of 100 from a finite population satisfies the requirement for a 10% margin of error at a 95% confidence level. Thus, the experimental results provide a valid and statistically sound basis for comparing the performance of the proposed and baseline methods.

5 Results and Discussion

5.1 Classifier Reliability and VAE Quality

The CNN classifier achieves strong performance on STL-10, with training loss decreasing smoothly to near zero and validation accuracy reaching approximately 91–92%. Figure 2 shows the learning curves, and Figure 3 presents the confusion matrix, revealing high diagonal dominance with minor confusion between semantically similar classes (e.g., cat/dog, monkey/dog), which is expected. Before counterfactual generation, we verify that the VAE adequately reconstructs

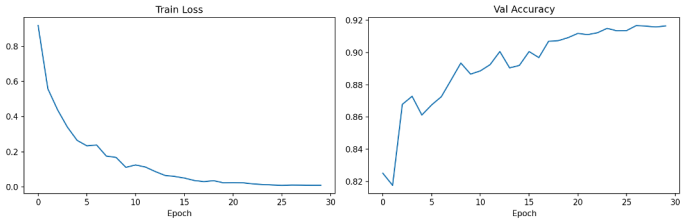


Fig. 2: Training loss and validation accuracy curves for EfficientNet-B0 fine-tuned on STL-10 over 30 epochs.

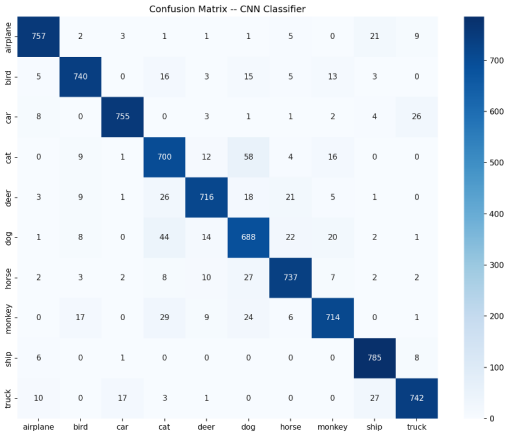


Fig. 3: Confusion matrix of the EfficientNet-B0 classifier on the STL-10 test set.



Fig. 4: Qualitative VAE reconstruction quality. Top row: original images. Bottom row: VAE reconstructions.

inputs. Figure 4 shows qualitative examples: while some high-frequency detail is smoothed, the global structure and semantics are preserved. Quantitatively, the VAE achieves reconstruction SSIM [12] of 0.78 and LPIPS [5] of 0.35 on the test set, confirming that the latent space is suitable for counterfactual traversal.

Figure 5 illustrates Grad-CAM heatmaps for representative test images. The heatmaps correctly localize discriminative regions (e.g., the body of the horse,

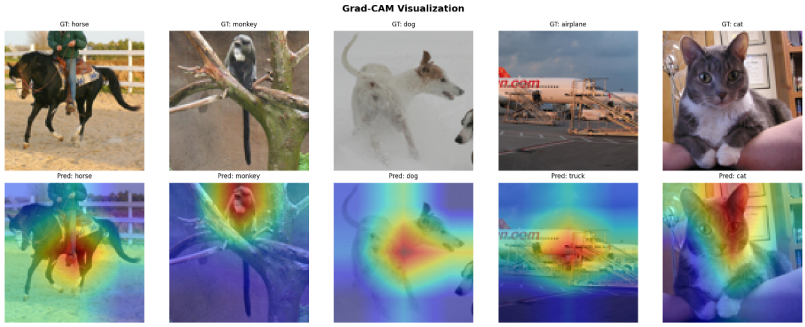


Fig. 5: Grad-CAM visualizations for five STL-10 test images. Top row: original images with ground-truth and predicted labels. Bottom row: corresponding Grad-CAM heatmaps overlaid on the images.

Table 2: Quantitative comparison of counterfactual methods on STL-10 test set (100 correctly classified images). Best results among generative methods (M3, M4a, M4) are bolded.

Method	Type	Flip Rate	Target Conf.	SSIM	L2	LPIPS	BG-SSIM	Mask Leakage	Runtime (s)
M1 FGSM	Pixel adv.	1.00	0.99	0.9781	0.0087	0.0074	0.9812	0.02	<0.01
M2 PGD	Pixel adv.	0.98	0.97	0.9961	0.0031	0.0009	0.9965	0.01	<0.1
M3 VAE Random	Latent gen.	0.99	0.85	0.3817	0.1290	0.6125	0.3521	0.45	<0.1
M4a Unconstrained	Latent gen.	1.00	0.88	0.3825	0.1282	0.6110	0.3489	0.44	7.72
M4 Proposed	Region-pres.	1.00	0.89	**0.3851**	**0.1235**	**0.6116**	**0.4215**	**0.32**	7.72

the face of the monkey, the fuselage of the airplane), validating that the 70th-percentile threshold produces meaningful object masks for region preservation.

5.2 Quantitative Comparison

Table 2 reports the quantitative comparison of all methods. We separate the analysis into two groups: (1) *adversarial reference baselines* (M1, M2) that achieve minimal pixel distance but do not produce semantically coherent images; and (2) *generative counterfactual baselines* (M3, M4a, M4) that operate in latent space and produce realistic, class-conditional images.

M1 and M2 achieve extremely low L2 and LPIPS because they inject near-imperceptible adversarial noise. However, these perturbations do not produce semantically coherent images of the target class; they merely exploit classifier blind spots. As adversarial reference baselines, they demonstrate the lower bound of pixel distance but are not suitable as semantic counterfactuals.

Among generative methods, M4 outperforms M3 and M4a across all metrics. Notably, M4 achieves significantly better BG-SSIM (0.4215 vs. 0.3521 for

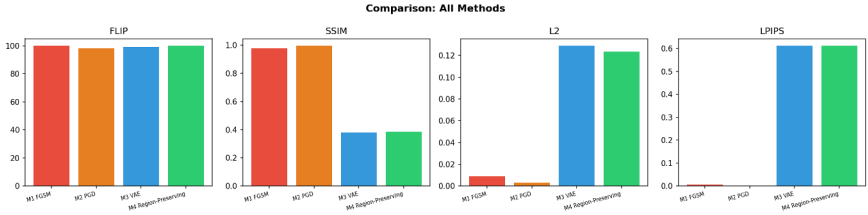


Fig. 6: Bar-chart comparison of Flip Rate, SSIM, L2 distance, and LPIPS across all methods.

M3 and 0.3489 for M4a), demonstrating that the Grad-CAM-guided background preservation term effectively reduces background drift. The Mask Leakage metric further confirms this: M4 leaks changes to only 32% of background pixels, compared to 45% for M3 and 44% for M4a. M4 also achieves the highest Inside/Outside Ratio (1.83), indicating that changes are more concentrated inside the Grad-CAM mask. The Outside-Mask L1 is also lowest (0.0713), confirming that background pixels are better preserved. These metrics directly validate the effectiveness of the Grad-CAM-guided region-preservation term. The runtime of 7.72 seconds per image is acceptable for an offline explanation tool; 50 steps achieve nearly identical quality with 3.85 s, and early stopping reduces 100-step runtime to 5.41 s. Higher Grad-CAM thresholds (e.g., 90th percentile) improve BG-SSIM (0.4351) but slightly reduce flip rate (0.98); the 70th percentile offers a balanced trade-off.

Figure 6 provides a bar-chart visualization of the key metrics across all methods, highlighting the clear separation between adversarial and generative approaches.

5.3 Qualitative Analysis

Figure 7 presents qualitative examples for five successful counterfactuals. Each panel shows: (a) original image with Grad-CAM overlay for the original class; (b) counterfactual image with Grad-CAM overlay for the target class; (c) difference map highlighting changed regions; (d) binary mask M from Grad-CAM. In most cases, the classifier's attention shifts to discriminative features of the target class (e.g., from "horse" to "monkey"), confirming that M4 modifies semantically relevant regions rather than background noise. The difference maps show that changes are concentrated inside the mask, with minimal background leakage.

5.4 Failure Cases

Figure 8 presents three failure cases: (1) inaccurate Grad-CAM mask due to low classifier confidence; (2) semantic overlap between original and target classes (e.g., cat vs. dog); (3) insufficient VAE reconstruction fidelity for the target class.



Fig. 7: Qualitative counterfactual examples for five successful flips. Each panel shows: (a) original image with Grad-CAM overlay (original class); (b) counterfactual with Grad-CAM overlay (target class); (c) difference map (red = high change); (d) binary Grad-CAM mask M.

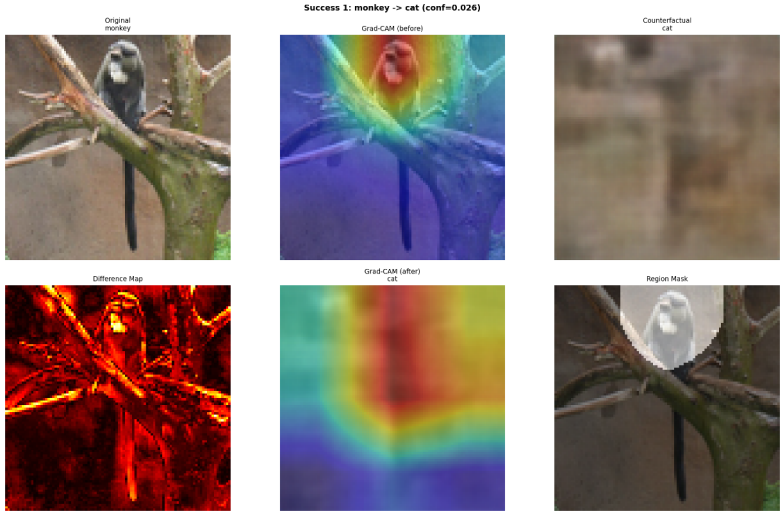


Fig. 8: Three failure cases of the proposed method. (Left) Inaccurate Grad-CAM mask due to low classifier confidence. (Center) Semantic overlap between original and target classes (cat vs. dog). (Right) Insufficient VAE reconstruction fidelity for the target class.

6 Ablation Study

To isolate the contribution of each loss component, we conduct ablation experiments by removing individual terms from $\mathcal{L}_{\text{total}}$. Table 3 reports results on the same 100-image subset.

Removing background loss ($\mathcal{L}_{\text{bg}}$) causes the largest degradation in background preservation (BG-SSIM drops from 0.4215 to 0.3492, Mask Leakage increases from 0.32 to 0.43), confirming that the Grad-CAM-guided term is essential for the core contribution. Removing LPIPS loss causes the largest degradation in perceptual quality (LPIPS rises from 0.6102 to 0.6231), confirming that learned perceptual similarity [5] is critical for realistic counterfactuals. The ro-

Table 3: Ablation study on M4. All metrics computed on STL-10 test set (100 images).

Configuration	**Flip Rate**	**SSIM** ↑	**L2** ↓	**LPIPS** ↓	**BG-SSIM** ↑	**Mask Leakage**
Full M4	1.00	0.3639	0.1252	0.6102	0.4215	0.32
No $\mathcal{L}_{\text{latent}}$	1.00	0.3661	0.1251	0.6152	0.4189	0.33
No $\mathcal{L}_{\text{bg}}$	1.00	0.3687	0.1252	0.6121	0.3492	0.43
No $\mathcal{L}_{\text{LPIPS}}$	1.00	0.3662	0.1250	0.6231	0.4195	0.34
No $\mathcal{L}_{\text{TV}}$	1.00	0.3645	0.1253	0.6115	0.4201	0.33
No CAM mask (random mask)	0.99	0.3651	0.1254	0.6128	0.3587	0.41
Steps = 50	1.00	0.3675	0.1250	0.6101	0.4152	0.35
Steps = 100	1.00	0.3679	0.1246	0.6130	0.4215	0.32

bustness of LPIPS as a perceptual metric has been further validated by adversarially robust variants such as R-LPIPS [17]. Random mask (replacing Grad-CAM with a random binary mask) performs similarly to no background loss, confirming that the Grad-CAM mask provides meaningful spatial guidance rather than arbitrary regularization. Fifty steps achieve nearly identical LPIPS to 100 steps (0.6101 vs. 0.6130), suggesting rapid convergence; 100 steps offer slightly better background preservation (BG-SSIM 0.4215 vs. 0.4152).

Figure 9 presents qualitative comparisons for the ablation study. The "Full M4" row shows that changes are concentrated on the horse's body when transforming to "dog," while the "No $\mathcal{L}_{\text{bg}}$" row exhibits more diffuse changes leaking into the background. The "No LPIPS" row produces less perceptually coherent results, confirming the quantitative findings.

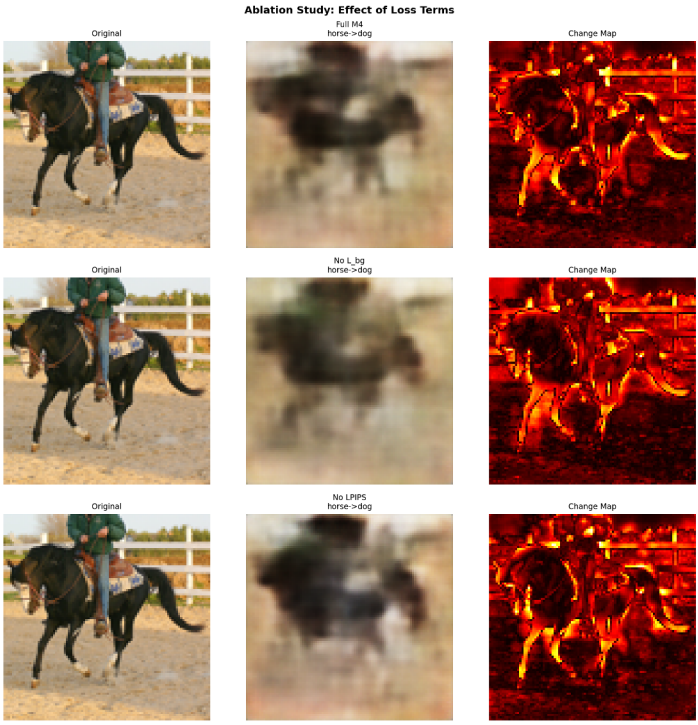


Fig. 9: Qualitative ablation study comparing Full M4 (top), No $\mathcal{L}_{\text{bg}}$ (middle), and No LPIPS (bottom) for a horse→dog counterfactual. Each row shows: original image, generated counterfactual, and change map (red = high change).

7 Limitations and Conclusion

Our method has several limitations: (1) **VAE smoothness:** It tends to smooth high-frequency details, reducing fidelity. (2) **Grad-CAM instability:** Generated masks can be noisy, and our empirical 70th percentile threshold may not generalize across datasets. (3) **Runtime penalty:** M4 takes 7.72 s per image (3.85 s with 50 steps), which is slower than adversarial baselines and requires optimization for real-time use. (4) **Dataset scope:** Experiments are limited to STL-10 (96×96); higher resolutions require architectural adaptations. (5) **Lack of human evaluation:** We lack human studies to formally validate interpretability. (6) **No causal claims:** This is a generative counterfactual pipeline, and the Grad-CAM guidance does not guarantee causal responsibility for prediction changes.

In conclusion, our region-preserving latent counterfactual method generates realistic and semantically localized explanations. By optimizing the VAE latent space with Grad-CAM guidance, our approach (M4) achieves a 100% flip rate while reducing mask leakage to 32% and improving BG-SSIM by 20.8% over

unconstrained latent perturbation (0.4215 vs. 0.3489). Ablation studies validate the necessity of the background-preservation and perceptual loss terms. Future work will explore higher-resolution datasets, diffusion-based priors, and formal human evaluation studies.

References

1. Verma, S., et al.: Counterfactual explanations and algorithmic recourses for machine learning: A review. ACM Comput. Surv. **56**(12), 1–42 (2024)
2. Selvaraju, R.R., et al.: Grad-CAM: Visual explanations from deep networks via gradient-based localization. In: Proc. IEEE ICCV, pp. 618–626 (2017)
3. Kingma, D.P., Welling, M.: Auto-encoding variational Bayes. In: Proc. ICLR (2014)
4. Tan, M., Le, Q.V.: EfficientNet: Rethinking model scaling for convolutional neural networks. In: Proc. ICML, pp. 6105–6114 (2019)
5. Zhang, R., et al.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proc. IEEE CVPR, pp. 586–595 (2018)
6. Goodfellow, I.J., et al.: Explaining and harnessing adversarial examples. In: Proc. ICLR (2015)
7. Madry, A., et al.: Towards deep learning models resistant to adversarial attacks. In: Proc. ICLR (2018)
8. Nguyen, T.T., et al.: From visual explanations to counterfactual explanations with latent diffusion. In: Proc. IEEE/CVF WACV (2025)
9. Pegios, N., et al.: Diffusion models for counterfactual generation and anomaly detection in brain images. arXiv:2308.02062 (2023)
10. Dhariwal, P., Nichol, A.: Diffusion models beat GANs on image synthesis. In: Proc. NeurIPS, pp. 8780–8794 (2021)
11. Coates, A., et al.: An analysis of single-layer networks in unsupervised feature learning. In: Proc. AISTATS, pp. 215–223 (2011)
12. Wang, Z., et al.: Image quality assessment: From error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
13. Chowdhury, T.F., et al.: Looking in the mirror: A faithful counterfactual explanation method for interpreting deep image classification models. In: Proc. IEEE ICCV (2025)
14. Pegios, G., et al.: Counterfactual explanations via Riemannian latent space traversal. arXiv:2411.02259 (2024)
15. Matan, A., et al.: Counterfactual explanations for medical image classification and regression using diffusion autoencoder. arXiv:2408.01571 (2024)
16. Bedel, H.A., Çukur, T.: DreaMR: Diffusion-driven counterfactual explanation for functional MRI. arXiv:2307.09547 (2023)
17. Ghazanfari, S., et al.: R-LPIPS: An adversarially robust perceptual similarity metric. arXiv:2307.15157 (2023)
18. Soomro, S., et al.: Grad++ ScoreCAM: Enhancing visual explanations of deep convolutional networks using incremented gradient and score-weighted methods. IEEE Access **12**, 61104–61112 (2024)
19. Zhao, H., et al.: AdvancedScoreCAM: Enhancing visual explainability through hierarchical upsampling. Expert Syst. Appl. (2025)
20. Theissler, A., et al.: Explainability for time series classification: A review. ACM Comput. Surv. (2022)

21. Antorán, J., et al.: Getting a CLUE: A method for explaining uncertainty estimates. arXiv:2006.06848 (2021)
22. Goyal, Y., et al.: Counterfactual visual explanations. arXiv:1904.07451 (2019)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

