Derivative-Free Diffusion Manifold-Constrained Gradient for Unified XAI

Supplementary Material

A. Proofs

Theorem 1. *Suppose that $\mathcal{M}$ is locally linear at $\boldsymbol{x}$, meaning that in the neighborhood of $\boldsymbol{x}$, the manifold aligns with its tangent space $\mathcal{T}_{\boldsymbol{x}}\mathcal{M}$. Suppose further, that the particles $\boldsymbol{x}^{(k)} \in \mathcal{M}$. Then, $\boldsymbol{g}_{\text{Free}}$ approximately lies within $\mathcal{M}$. Precisely, $\boldsymbol{C}_{\boldsymbol{xx}}$ acts as a linear transformation, expanding along $\mathcal{T}_{\boldsymbol{x}}\mathcal{M}$ and contracting the vectors normal to $\mathcal{T}_{\boldsymbol{x}}\mathcal{M}$.*

Proof. All particles $\boldsymbol{x}^{(k)}$ lie on $\mathcal{M}$ and are close to $\boldsymbol{x}$, so the vectors $\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}}$ approximately lie in $\mathcal{T}_{\boldsymbol{x}}\mathcal{M}$. For $\boldsymbol{b} \in \mathbb{R}^d$, we have

$$\begin{aligned}
\boldsymbol{C}_{\boldsymbol{xx}} \cdot \boldsymbol{b} &= \left(\frac{1}{K} \sum_{k=1}^{K} (\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})(\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})^\top \right) \cdot \boldsymbol{b} \\
&= \frac{1}{K} \sum_{k=1}^{K} (\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}}) \left((\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})^\top \cdot \boldsymbol{b} \right) \\
&= \frac{1}{K} \sum_{k=1}^{K} \left((\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})^\top \cdot \boldsymbol{b} \right) (\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}}) \in \mathcal{T}_{\boldsymbol{x}}\mathcal{M}.
\end{aligned}$$

Hence, for any vector $\boldsymbol{b} \in \mathbb{R}^d$, the multiplication of $\boldsymbol{C}_{\boldsymbol{xx}}$ on any $\boldsymbol{b}$ is on manifold $\mathcal{M}$. More precisely, consider the eigen decomposition of $\boldsymbol{C}_{\boldsymbol{xx}} = \sum_{i=1}^{d} \lambda_i \mathbf{e}_i \mathbf{e}_i^\top$:

$$\begin{aligned}
\boldsymbol{C}_{\boldsymbol{xx}} \cdot \nabla_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) &= \left(\sum_{i=1}^{d} \lambda_i \mathbf{e}_i \mathbf{e}_i^\top \right) \cdot \nabla_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) \\
&= \sum_{i=1}^{d} \lambda_i \mathbf{e}_i \left(\mathbf{e}_i^\top \cdot \nabla_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) \right) \\
&= \sum_{i=1}^{d} \lambda_i \left(\mathbf{e}_i^\top \cdot \nabla_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) \right) \mathbf{e}_i \\
&\approx \sum_{i=1}^{m} \lambda_i \left(\mathbf{e}_i^\top \cdot \nabla_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) \right) \mathbf{e}_i \in \mathcal{T}_{\boldsymbol{x}}\mathcal{M}
\end{aligned}$$

In this approximation, the summation is limited to m terms, with m being the intrinsic dimension of $\mathcal{T}_{\boldsymbol{x}}\mathcal{M}$ (and $\mathcal{M}$). Thus, $\nabla_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x})$ is scaled along the principal directions of the particles and contracted along the direction normal to the data manifold. □

Theorem 2. *Suppose $\|\nabla_{\boldsymbol{x}}^2 \boldsymbol{f}(\boldsymbol{x})\|$ be uniformly bounded and there exists $\delta > 0$ such that $\boldsymbol{x}^{(k)} \in \{\boldsymbol{x} \in \mathbb{R}^d : \|\boldsymbol{x} - \bar{\boldsymbol{x}}\| \leq \delta\}$ for all k. Then, the following approximation holds*

$$\boldsymbol{C}_{\boldsymbol{xf}} = \boldsymbol{C}_{\boldsymbol{xx}} \nabla_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x})^\top + \mathcal{O}(\delta^3), \tag{9}$$

where $\boldsymbol{C}_{\boldsymbol{xf}} := \frac{1}{K} \sum_{k=1}^{K} (\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})(\boldsymbol{f}(\boldsymbol{x}^{(k)}) - \bar{\boldsymbol{f}})^\top$ and $\bar{\boldsymbol{f}} := \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{f}(\boldsymbol{x}^{(k)})$.

Proof.

$$\boldsymbol{C}_{\boldsymbol{xf}} = \frac{1}{K}\sum_{k=1}^{K}(\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})(\boldsymbol{f}(\boldsymbol{x}^{(k)}) - \bar{\boldsymbol{f}})^\top \tag{15}$$

$$= \frac{1}{K}\sum_{k=1}^{K}(\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})(\boldsymbol{f}(\boldsymbol{x}^{(k)}) - \boldsymbol{f}(\bar{\boldsymbol{x}}) + \boldsymbol{f}(\bar{\boldsymbol{x}}) - \bar{\boldsymbol{f}})^\top \tag{16}$$

$$\overset{(a)}{=} \frac{1}{K}\sum_{k=1}^{K}(\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})\left(\left(\frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right)_{\boldsymbol{x}=\bar{\boldsymbol{x}}}(\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}}) + \mathcal{O}(\delta^2) + [\boldsymbol{f}(\bar{\boldsymbol{x}}) - \bar{\boldsymbol{f}}]\right)^\top \tag{17}$$

$$= \frac{1}{K}\sum_{k=1}^{K}(\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})\left(\left(\frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right)_{\bar{\boldsymbol{x}}}(\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})\right)^\top + \mathcal{O}(\delta^3) + \frac{1}{K}\sum_{k=1}^{K}(\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})\left(\underbrace{\boldsymbol{f}(\bar{\boldsymbol{x}}) - \bar{\boldsymbol{f}}}_{\text{const}}\right)^\top$$

$$\overset{(b)}{=} \frac{1}{K}\sum_{k=1}^{K}(\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})(\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}})^\top\left(\frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right)_{\bar{\boldsymbol{x}}}^\top + \mathcal{O}(\delta^3) \tag{18}$$

$$= \boldsymbol{C}_{\boldsymbol{xx}}\nabla_{\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x})^\top + \mathcal{O}(\delta^3) \tag{19}$$

where (a) is given by the Taylor expansion

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{f}(\bar{\boldsymbol{x}}) + \nabla_{\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x})|_{\bar{\boldsymbol{x}}}(\boldsymbol{x} - \bar{\boldsymbol{x}}) + \mathcal{O}(\|\nabla^2\boldsymbol{f}\| \cdot \|\boldsymbol{x} - \bar{\boldsymbol{x}}\|^2)$$

and (b) is from $\frac{1}{K}\sum_{k=1}^{K}(\boldsymbol{x}^{(k)} - \bar{\boldsymbol{x}}) = 0$. □

B. Chest X-ray Counterfactual Generation

For CXR counterfactual generation, we empirically find that using direct gradient-ascent with FreeMCG (i.e., Eq. 13), rather than reverse diffusion as was done with ImageNet, produces more realistic results. Details in Appendices G and H.

B.1. CXR Counterfactuals: Disease → Normal

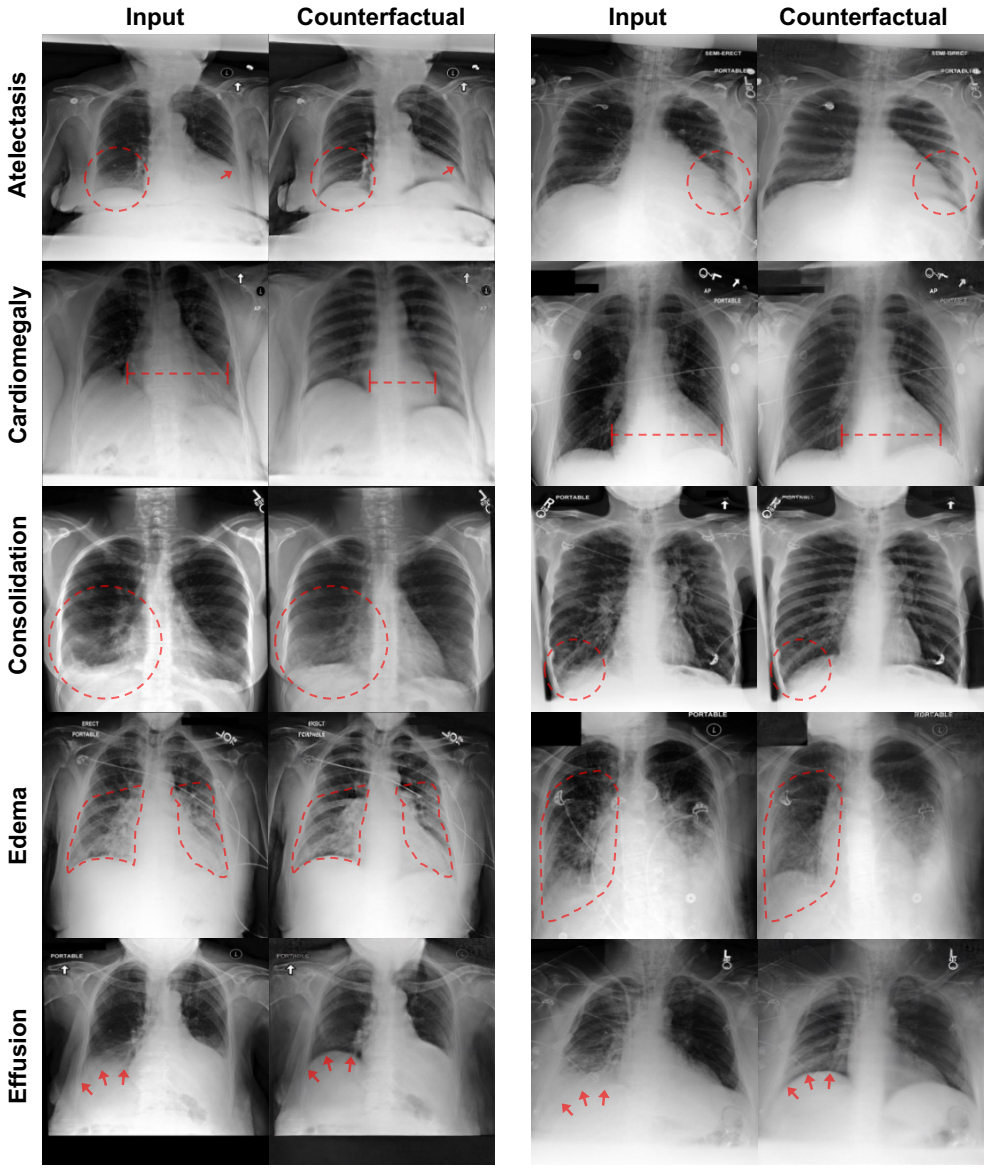

Figure 6. Disease→Normal class counterfactuals produced using FreeMCG for each type of disease lesion.

In Disease→Normal counterfactuals (Fig. 6), we observe that the features associated with each lesion are removed or decreased in the generated counterfactuals. For example, the `cardiomegaly`→`normal` counterfactual shows decreased heart size and `effusion` → `normal` counterfactual shows sharper costophrenic angles, corresponding to decreased appearance of effusion. Note that for cardiomegaly, the generated counterfactuals also make the ribs more distinct. This tells the human viewer that the classification model has learned to associate more visible ribs with normal CXRs without cardiomegaly. This is a type of spurious correlation arising from the distribution of training data (MIMIC-CXR) used to train the model and serves as an example of how counterfactual generation gives visual insight about the decision boundary of the model that is not obtainable from other forms of XAI techniques such as feature attribution.

B.2. CXR Counterfactuals: Normal → Disease

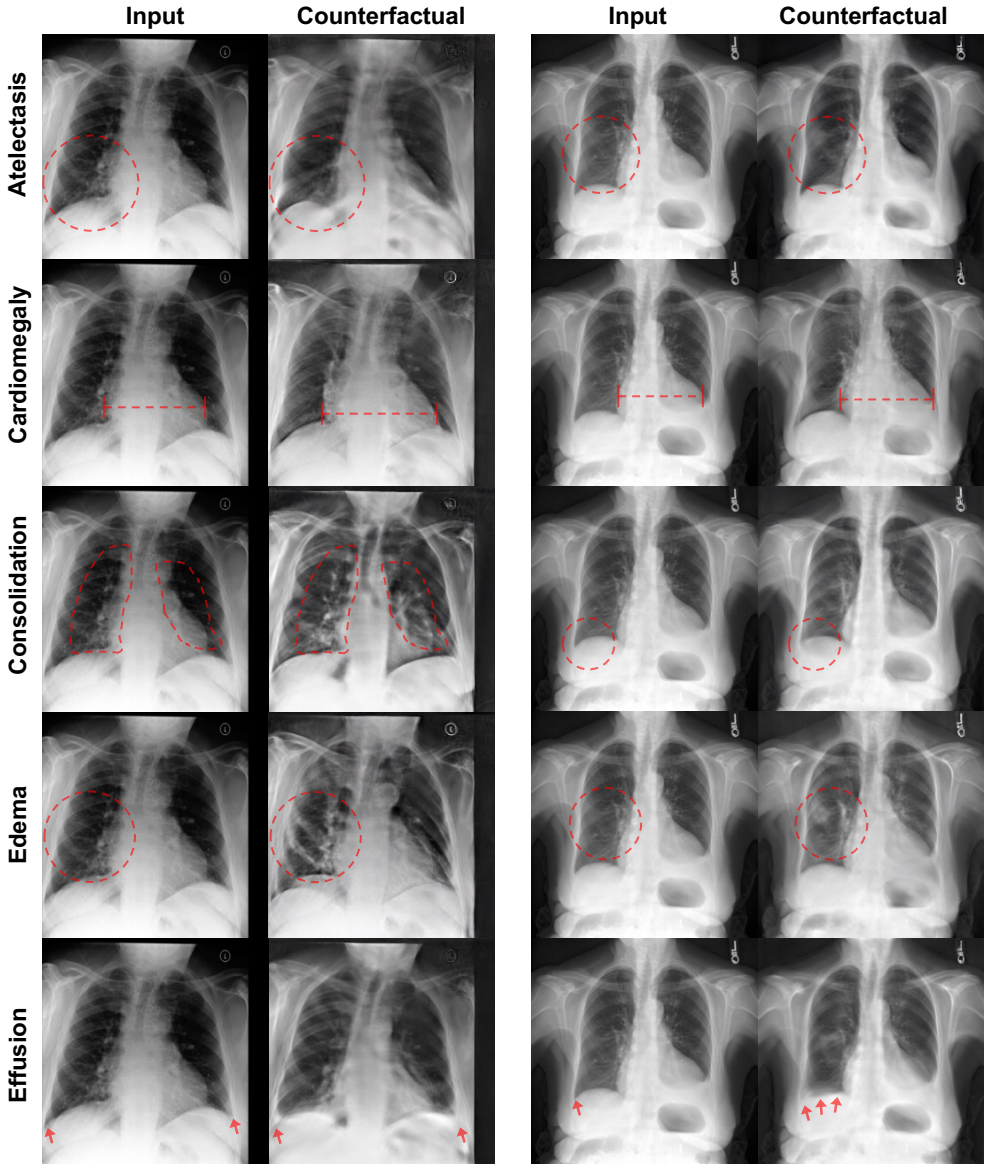

Figure 7. Normal→Disease counterfactuals for each type of disease lesion.

From Normal→Disease counterfactuals (Fig. 9), we can gain a visual understanding of what features the model associates with each lesion *e.g.* larger cardiac diameter for cardiomegaly, blunted costophrenic angle for effusion. Note that in the example in the second column, we can see that the `normal→consolidation` counterfactual generation elicits features more similar to effusion rather than consolidation itself. This tells us that the model associates presence of effusion with consolidation.

B.3. CXR Counterfactuals: Comparison with DVCE

Table 2 includes the quantitative evaluation results for CXR counterfactual generation. Because LDCE [16] requires a separately pretrained text-to-image diffusion model, we only compare CXR counterfactual results with the DVCE [4] baseline – again without the cone projection to adversarially robust weights as we are assuming lack of access to any adversarially trained model weights. We look at the Disease→Normal direction counterfactual generation as we anticipate this to be the most common real usage scenario.

	Flip Rate ↑	L2 ↓	FID ↓
DVCE [4]	97.9	**526.17**	**23.57**
FreeMCG	**99.2**	1367.07	42.54

Table 2. Quantitative (Flip Rate, L2, FID) and human user study (perceived change, perceived similarity, perceived realism) results.

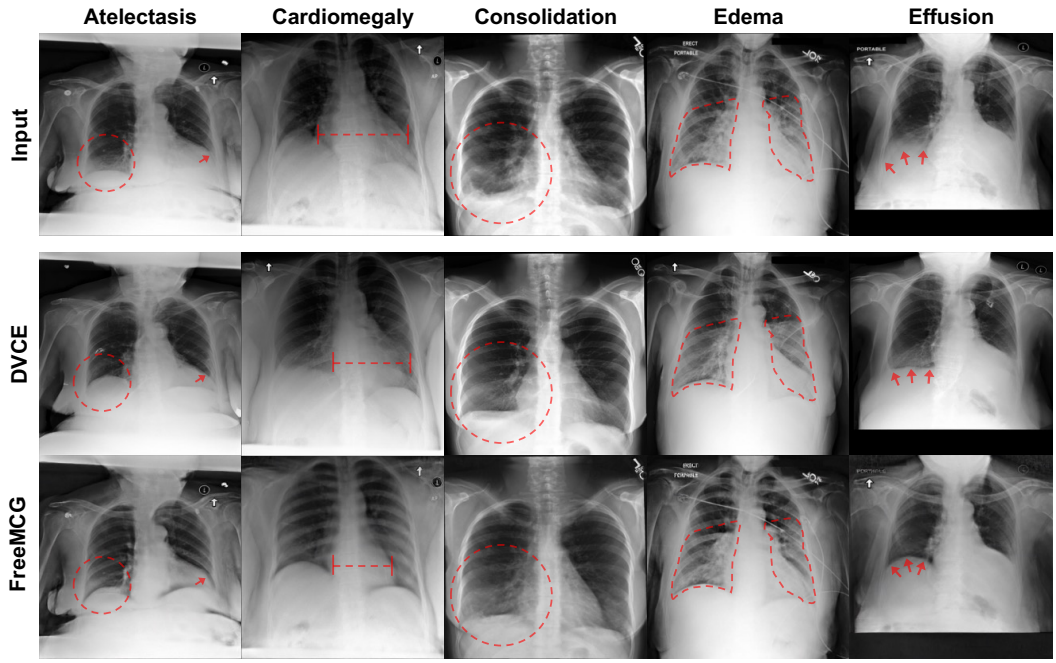

Figure 8. Disease→Normal counterfactuals for each lesion using DVCE vs. FreeMCG.

Both DVCE and FreeMCG produce counterfactuals that reduce the lesion features and move the CXR image towards the normal class, but FreeMCG produces more sparse changes than DVCE (Fig. 8). Compared to FreeMCG, DVCE more often produces outputs closer to adversarial attacks, (prediction of the classifier is flipped but changes made to the image are not very significant) resulting in lower L2 and FID. It is notable that FreeMCG, despite being a gradient-free method, achieves higher flip rate than the real gradient-based DVCE (Tab. 2).

C. Counterfactuals for different architectures

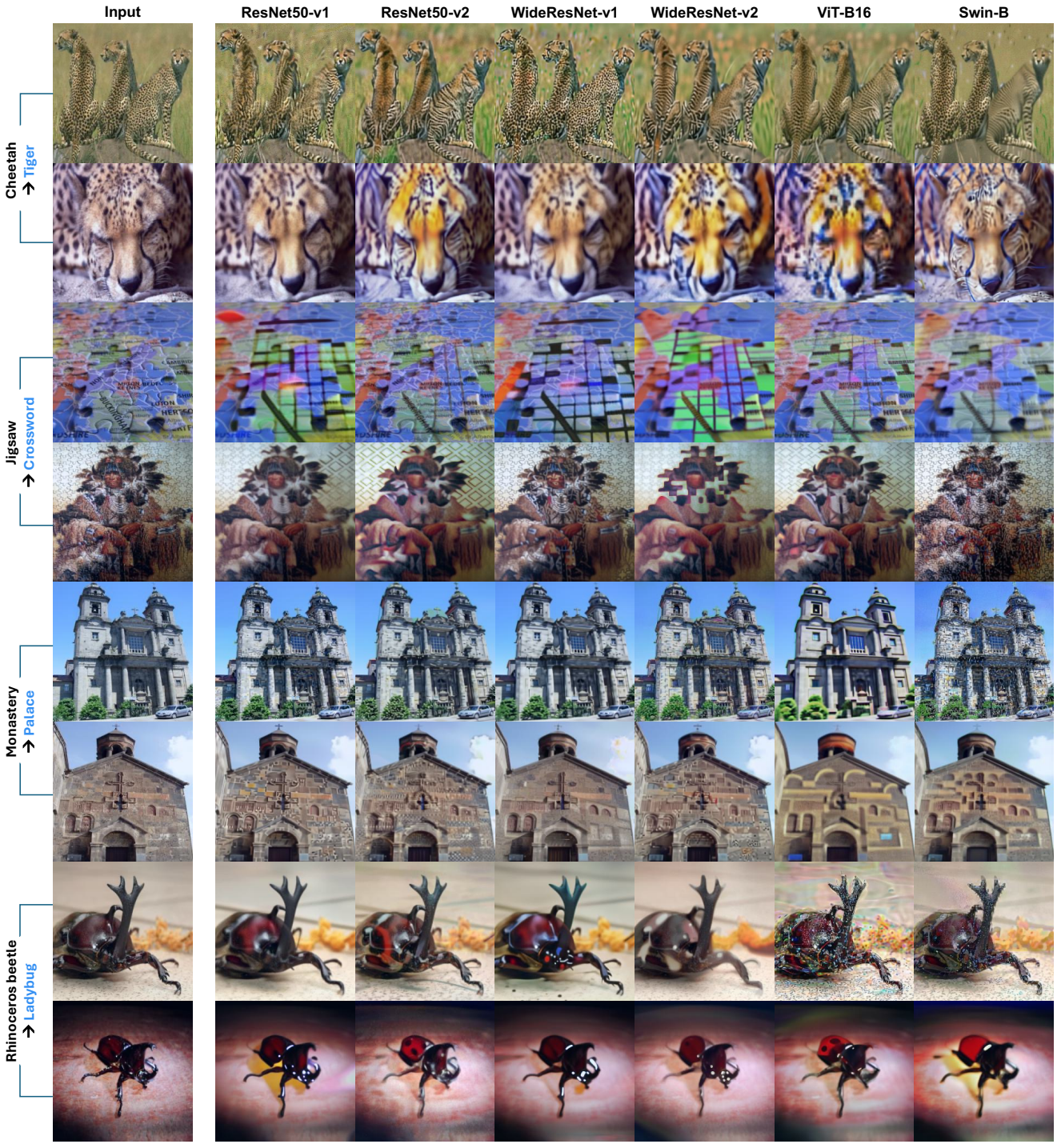

Figure 9. Illustration of decision boundaries of different architectures using FreeMCG-based counterfactual generation. All models are from `torchvision.models`, and `v1` and `v2` refer to different weights trained using different preprocessing pipelines. Both preprocessing and model architecture affect the decision boundary. For example, for cheetah $\rightarrow$ tiger counterfactual generation, classifiers trained using `v2` preprocessing seem to have decision boundaries more in line with human perception. In addition, certain architectures show poorly defined decision boundaries for certain input images. This type of comparison can be used to determine which models are more human-aligned for a given input image.

D. FreeMCG Meets Latent Diffusion

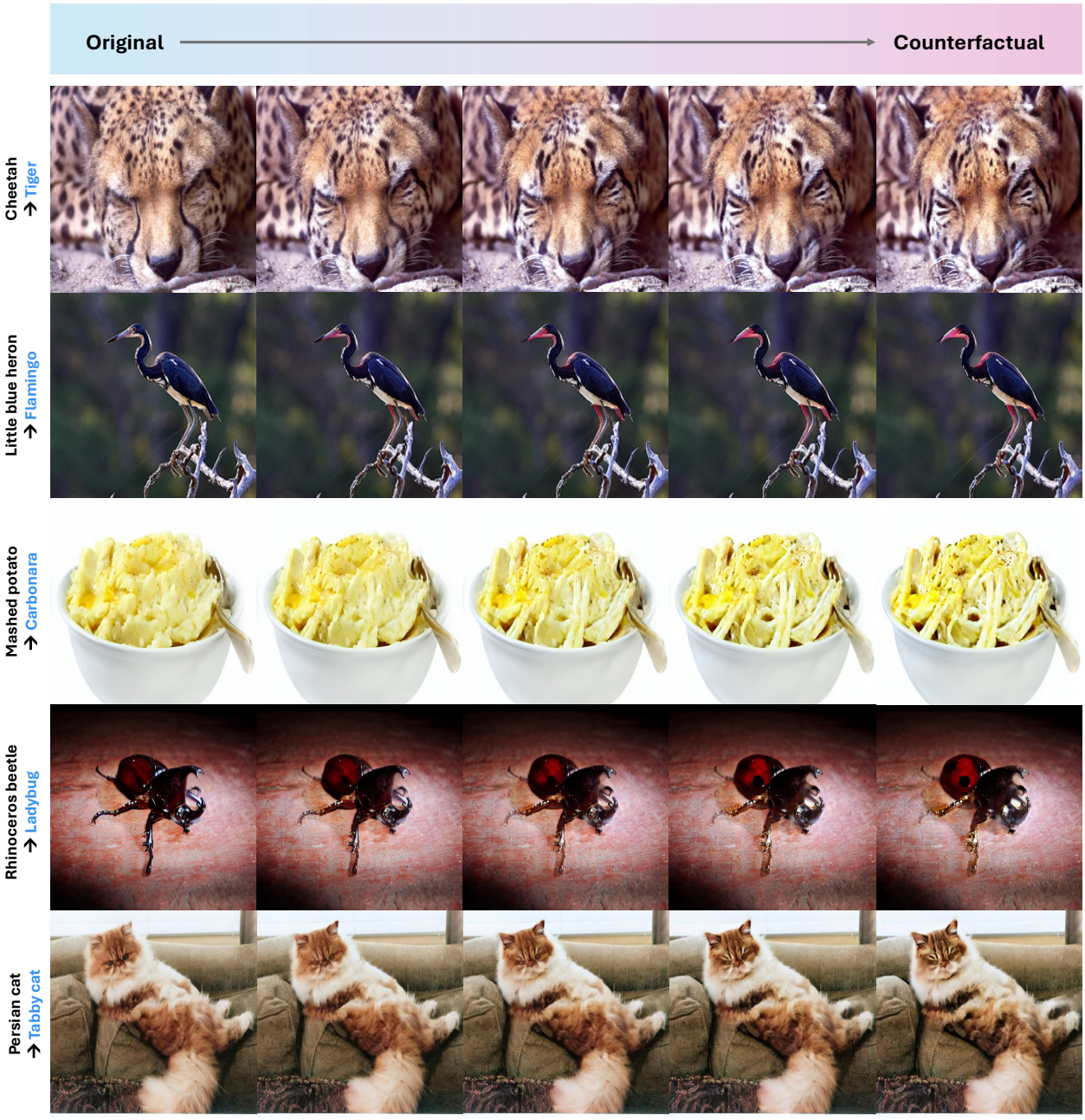

Figure 10. FreeMCG-counterfactual generation using Stable Diffusion [41] with direct FreeMCG-ascent in the latent space. Leftmost column shows the input images, and rightmost column shows the final generated counterfactuals for the target class. The columns in between show intermediate steps during generation.

FreeMCG can be analogously applied in the latent space to generate counterfactuals using pretrained latent diffusion models such as Stable Diffusion [41]. Note that FreeMCG **does not use any text-conditioning** and uses only unconditional sampling. In LDCE [16], text conditions such as `'a photo of a tiger'` are used during generation; while this assists in the generation process, it may introduce biases of the generative model and risks reducing the faithfulness of the explanation as the generated output does not depend solely on the classification model of interest.

E. Additional Feature Attribution Results

E.1. Additional feature attribution baselines

Figure 11 shows results for gradient-based methods (e.g., vanilla gradient [50], Integrated Gradients [49], InputXGradient [47]) and perturbation/removal-based methods (e.g. SHAP [31], HSIC [36], RISE [38]).

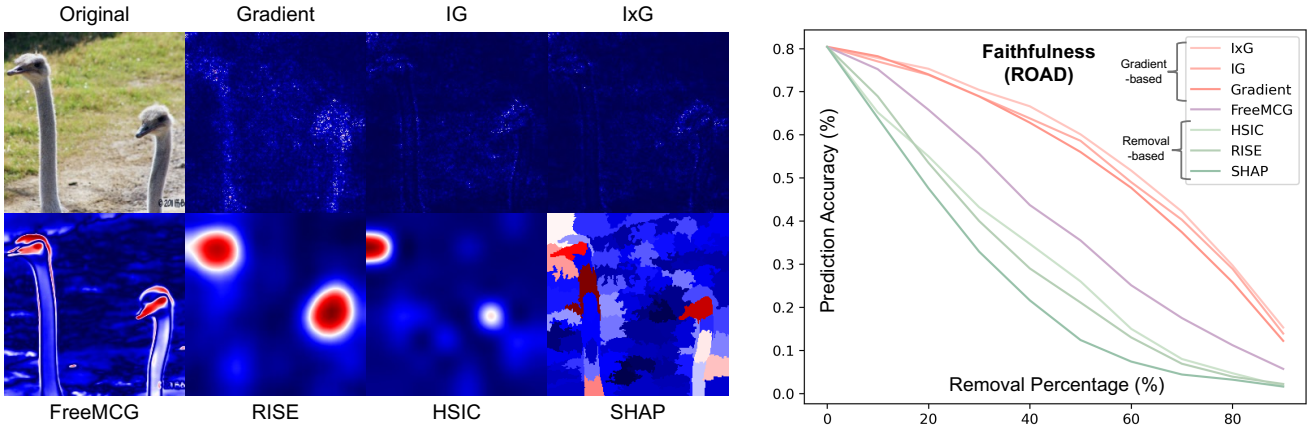

Figure 11. Feature attributions from each method (Left) and their ROAD results (Right)

Perturbation-based methods such as SHAP, HSIC, and RISE are black-box feature attributions that share the common theme of removing certain features (pixels) in the image and attributing importance values by observing how much change is observed in the prediction of the model being explained. Because they are removal-based, these methods naturally perform better on Faithfulness metrics such as ROAD 11; however, these methods are unable to estimate the gradient direction – and therefore do not contain information that can be used for other applications such as counterfactual generation. FreeMCG, while estimating the gradient, shows faithfulness (based on ROAD) closer to that of these removal-based methods.

E.2. Additional metrics for feature attribution quality: Sensitivity & Complexity

Besides **faithfulness** (i.e., how *faithfully* the explanation reflects the decision of the model being explained), the other major desiderata for feature attribution explanations are **sensitivity** and **complexity** [5]. The **sensitivity** criterion refers to the desideratum that good explanations should remain stable to minor perturbations to the input while the **complexity** criterion to the fact that less complex (i.e., lower entropy) explanations are more interpretable and useful to humans. We measure sensitivity and complexity as explained in Bhatt et al. [5] and report the results in Table 3.

	Grad.	IG	IxG	RISE	HSIC	SHAP	FreeMCG (**ours**)
Avg. Sens. (↓)	1.09	1.32	1.43	0.68	2.44	9.11	**0.40**
Max Sens. (↓)	2.16	2.51	2.80	0.71	4.05	17.16	**0.60**
Compl. (↓)	10.40	10.08	**10.05**	10.53	10.54	10.58	10.41

Table 3. Sensitivity & Complexity of feature attributions

FreeMCG has the lowest (best) sensitivity by far while there is no meaningful difference in complexity across different methods.

F. Ablation Studies

We test the effect of varying (1) the particle count (i.e., the number of perturbations $x_{0|t}^{(k)}$ used to compute the FreeMCG gradient estimation), (2) the diffusion timestep used to begin the reverse diffusion process, and (3) the proximity regularization coefficient (Table 4).

	# particles				diffusion time t'				prox. reg. β			
	10	50	100	200	0.3	0.4	0.5	0.6	0.00	0.01	0.02	0.03
Flip rate (↑)	42.0	45.0	**46.0**	**46.0**	40.0	**46.0**	42.0	36.0	**65.0**	59.0	46.0	28.0
L2 (↓)	472.0	345.1	345.1	**310.1**	**343.0**	345.1	348.9	348.3	593.7	408.5	345.1	**300.1**
FID (↓)	160.3	135.5	**133.3**	136.4	**131.0**	133.3	138.1	146.2	250.7	177.3	133.3	**100.8**

Table 4. Ablation study on hyperparameters of FreeMCG (100 images used per ablation).

We observe that higher particle count improves results, but this plageaus at 100 particles; timestep of $0.4 * T$ shows highest conversion to the target class; and proximity regularization (β) serves its intended purpose of keeping the generated counterfactual close to the original image (i.e., $\beta \uparrow \rightarrow$ Flip $\downarrow$, L2 $\downarrow$, FID $\downarrow$).

G. Implementation details

For all experiments including feature attribution and counterfactual explanation, we use 100 particles per each timestep, which is much smaller than $\geq 10,000$ particles that are used in Zheng et al. [62].

Feature attribution To save computation while taking advantage of the full scale space of the pretrained diffusion model across timesteps, we choose $t \in [100, 200, 300, 400, 500, 600, 700]$ with 100 particles per timestep to compute FreeMCG. The average across color channels is used as the feature attribution.

Counterfactual explanation For ImageNet counterfactual generation, we use the the hyperparameters $t' = 400$ and $\alpha = 0.2$. For β, we experiment with both 0.01 and 0.02 and select the most informative example. To stabilize gradients, we employ $\gamma_t := \sqrt{\bar{\alpha}_t \bar{\alpha}_{t-1}}$ as proposed in Song et al. [53]. Further, we utilize the normalized version of the gradients as used in Augustin et al. [4]. We use DDIM [52] sampling with 100 steps. For MIMIC-CXR counterfactual generation, we use FreeMCG directly for gradient ascent on the given input image as we find empirically that this produces more realistic results for CXR data. We use 18 iterations of gradient ascent steps at $t' = 300, \alpha = 0.2$.

H. Experiment Details

ImageNet classifier We use the pretrained ImageNet classification models provided by `torchvision.models`[3]. Results in the main paper are obtained using the ResNet50 [20] architecture and `ResNet50_Weights.IMAGENET1K_V2` weights from `torchvision.models`.

ImageNet diffusion model We use the pretrained unconditional diffusion model publicly available at the `openai/guided-diffusion` [13] repository[4]. Note that one should use unconditional diffusion (rather than conditional diffusion) for counterfactual generation, as the direction of generation should depend only on the model of interest for it to be a faithful explanation of that model.

MIMIC-CXR classifier We train a chest X-ray classification model on the ResNet50 architecture [20] using the MIMIC dataset [26] on the five CheXpert competition labels (atelectasis, cardiomegaly, consolidation, edema, pleural effusion) along with the 'normal' class.

MIMIC-CXR diffusion model We train an unconditional diffusion model on MIMIC CXR images using the same recipe as the diffusion model used for ImageNet [13]; but because CXR images are much less diverse than ImageNet, we decrease the size of the U-net to have 128 channels and attention resolutions of 16, 8 (compared to 256 channels and attention resolutions of 32, 16, 8 for ImageNet diffusion U-net). In addition, since CXR images are greyscale images (i.e., not RGB) and therefore require only one color channel, we experimented with both 1-channel and 3-channel diffusion and found that the results were consistent with no discernible differences.

ROAD [42] for CXR feature attribution As there are only six classes for chest X-ray classification, rather than reporting the drop in accuracy (which were not very informative due to the fact that CXR classifiers are less accurate than ImageNet classifiers to start with and there is a relatively high chance of "guessing" the correct answer due to the low number of classes), we measure the drop in the predicted probability for the originally predicted class. This assessment operates on the same principle as the ROAD for ImageNet (i.e., model prediction decline with feature removal) but allows for a more fine-grained assessment.

[3] https://pytorch.org/vision/main/models.html
[4] https://github.com/openai/guided-diffusion

I. Human User Study for Generated Counterfactuals

For human user assessment, we conducted an anonymized user study through Amazon Mechanical Turk and recruited 158 participants with diverse age, gender, and AI familiarty backgrounds (Fig. 12). We asked each user to assess generated counterfactuals on three different categories: **(1) change** towards target, **(2) similarity** to input, and **(3) realism** of the generated image by ranking counterfactuals generated using FreeMCG, DVCE, LDCE across the three categories.

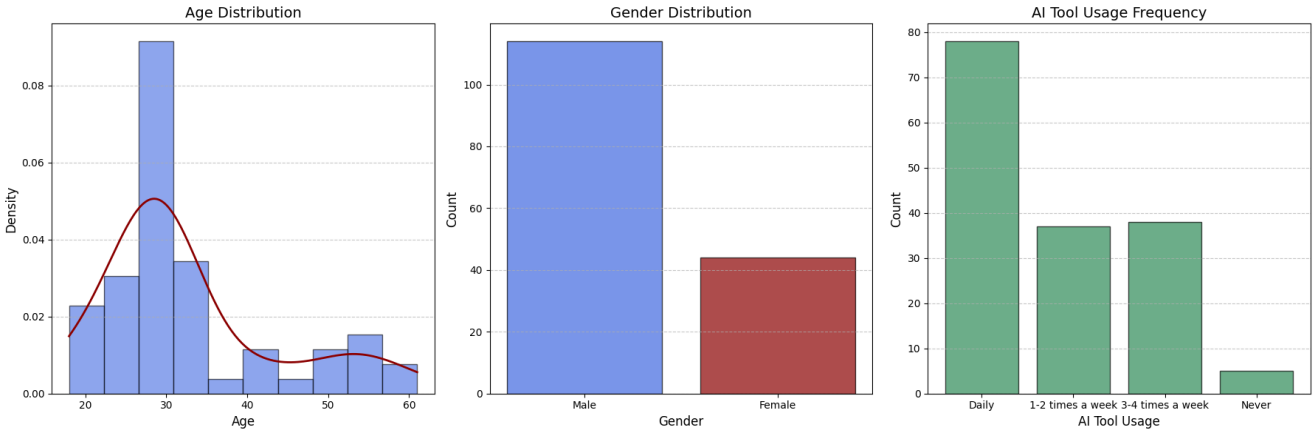

Figure 12. User study demographic statistics