Diffusion Models for Attribution

Xiongren Chen[1], Jiuyong Li[1], Jixue Liu[1], Lin Liu[1], Stefan Peters[1], Thuc Duy Le[1], Wentao Gao[1], Xiaojing Du[1], Anthony Walsh[2]

[1]University of South Australia
[2]Green Triangle Forest Industries Hub
{xiongren.chen, wentao.gao, xiaojing.du}@mymail.unisa.edu.au,{jiuyong.li,jixue.liu,lin.liu, stefan.peters,thuc.le}@unisa.edu.au,anthony@gtfih.com.au

Abstract

In high-stakes domains such as healthcare, finance, and law, the need for explainable AI is critical. Traditional methods for generating attribution maps, including white-box approaches relying on gradients and black-box techniques that perturb inputs, face challenges like gradient vanishing, blurred attributions, and computational inefficiencies. To overcome these limitations, we introduce a novel approach that leverages diffusion models within the framework of information bottleneck theory. By utilizing the Gaussian noise from diffusion models, we connect the information bottleneck with the Minimum Mean Squared Error (MMSE) from classical information theory, enabling precise calculation of mutual information. This connection leads to a new loss function that minimizes the Signal-to-Noise Ratio (SNR), facilitating efficient optimization and producing high-resolution, pixel-level attribution maps. Our method achieves greater clarity and accuracy in attributions than existing techniques, requiring significantly fewer pixel values to reach the necessary predictive confidence. This work demonstrates the power of diffusion models in advancing explainable AI, particularly in identifying critical input features with high precision.

Introduction

As machine learning continues to evolve and expand its applications, explainability of these systems becomes increasingly critical, especially in high-stakes areas like healthcare (Chaddad et al. 2023; Saraswat et al. 2022), finance (Soundararajan and Shenbagaraman 2024) and judiciary (Gless 2019). Various explainable machine learning methods have been developed, and they generally fall into two categories: white-box approach and black-box approach (Novello, Fel, and Vigouroux 2022; Chen et al. 2024).

White-box and black-box methods each have distinct approaches and limitations in generating attribution maps for explainability. White-box methods, which use gradients from the input layer (Simonyan, Vedaldi, and Zisserman 2014; Smilkov et al. 2017; Sundararajan, Taly, and Yan 2017; Pan, Li, and Zhu 2021) or map importance from the representation space back to the input (Schulz et al. 2020; Zhang et al. 2021), can suffer from issues like gradient vanishing or exploding in deep networks, leading to inaccurate interpretations (Sundararajan, Taly, and Yan 2017; Hanin 2018), and blurred attribution maps due to information loss when mapping from the representation space to pixel space. In contrast, black-box methods, which directly perturb the input images and analyze the changes in the model's output (Petsiuk, Das, and Saenko 2018; Yang et al. 2021; Novello, Fel, and Vigouroux 2022; Chen et al. 2024), can theoretically produce clearer and more precise attribution maps. However, this approach is computationally expensive when perturbing individual pixels, and while methods like HSIC-Attribution (Novello, Fel, and Vigouroux 2022) and SMDL-Attribution (Chen et al. 2024) mitigate this by perturbing regions of pixels, they do so at the cost of reduced resolution and precision in the resulting attribution maps.

In this paper, we reformulate attribution as an information bottlenecks (Tishby, Pereira, and Bialek 2000) optimization problem by introducing noise perturbations directly on individual pixels, rather than on representation features, to address the three key limitations of existing white-box and black-box methods: (1) reliance on gradients, (2) mapping attributions from the representation space to the input pixel space, and (3) perturbing individual pixels. However, optimizing information bottlenecks in the pixel space presents a significant challenge: calculating mutual information from a data distribution perspective to determine pixel importance is impractical because the pixel space distribution is unknown. Mutual information computation becomes intractable with high-dimensional data and often depends on assumptions—such as those in the IBA method (Schulz et al. 2020) that assume features follow a normal distribution—assumptions that do not hold in pixel space.

To address the challenge, we propose using diffusion models (Luo 2022) as information bottlenecks. By leveraging the Gaussian noise in diffusion models, we connect the information bottleneck with the MMSE from classical information theory (Guo, Shamai, and Verdú 2005). This approach leads to an integral expression for mutual information, and we demonstrate that reducing the SNR effectively decreases mutual information. Building on this, we design a new loss function that directly minimizes SNR, simplifying computations within the information bottleneck framework. Ultimately, our method achieves pixel-level attribution in the input layer by utilizing diffusion models and the integral form of mutual information. Figure 1 visually compares the

attribution maps generated by our method with those from other white-box and black-box methods. Both the white-box method IBA and the black-box method HSIC-Attribution have limitations that produce blurred, low-resolution attribution maps. In contrast, our method generates high-resolution attribution maps that more precisely capture fine-grained details and critical features in the input image.

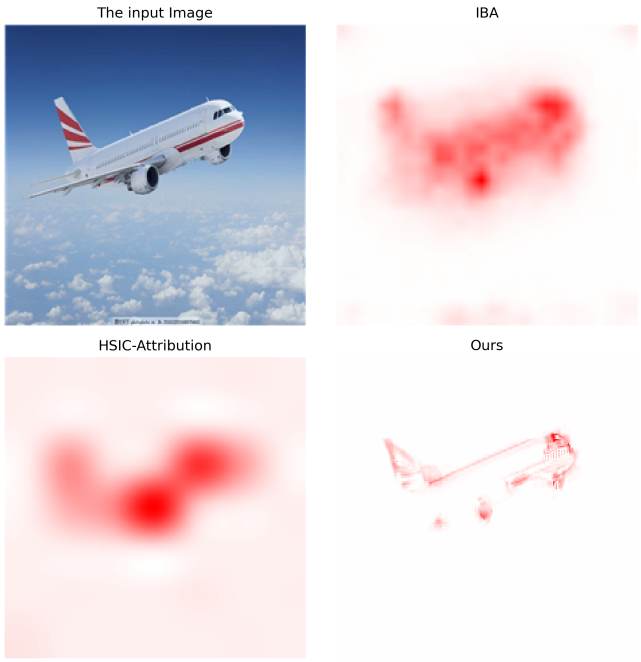

Figure 1: Existing white-box methods like IBA often produce blurred attribution maps, while black-box methods such as HSIC-Attribution generate low-resolution maps. In contrast, our proposed pixel-level attribution method delivers clearer and more precise attribution maps.

To summarize our contributions:

- By integrating diffusion models into information bottleneck theory, we significantly simplified the optimization framework. We demonstrated that reducing mutual information is equivalent to minimizing the SNR. This innovative derivation simplifies the application of information bottleneck theory, substantially reducing model complexity and addressing computational challenges associated with high-dimensional data, such as high-resolution images.
- We derived and presented an integral formula for directly calculating mutual information, along with an algorithmic implementation using diffusion models. This approach not only enables precise mutual information computation but also reliably quantifies the effects of the information bottleneck, providing a solid theoretical foundation for model optimization. Unlike traditional methods that rely on approximations and assumptions (Schulz et al. 2020), our method significantly improves the accuracy and efficiency of these calculations.
- Our method demonstrates superior efficiency and precision, achieving the required predictive confidence with less than 3% of pixel values, compared to at least 13% required by other methods. This makes a significant breakthrough in identifying the most critical input information.

Related Work

In this section, we review various explainable AI methods, categorized into Gradient-based Methods, Backpropagation Techniques, CAM-based Approaches, Information Theory Approaches, and Perturbation-based Methods. We also classify and label these methods as white-box or black-box.

Gradient-based Methods (white-box): Saliency Maps (SM) (Simonyan, Vedaldi, and Zisserman 2014) introduced gradients as importance scores for visual explanations but struggled with noise and instability. SmoothGrad (Smilkov et al. 2017) mitigated this by averaging maps. Integrated Gradients (Sundararajan, Taly, and Yan 2017) and AGI (Pan, Li, and Zhu 2021) further improved accuracy but faced computational challenges.

Backpropagation Techniques (white-box): Guided Backpropagation (Guided-BP) (Springenberg et al. 2014) refines this by removing negative gradients, producing pixel-level heatmaps. However, these methods can reconstruct entire images without clearly indicating key areas (Adebayo et al. 2018). Layer-wise Relevance Propagation (LRP) (Bach et al. 2015) and DeepLIFT (Shrikumar, Greenside, and Kundaje 2017) enhance clarity by backpropagating relevance and adding reference points.

CAM-based Approaches (white-box): Class Activation Map (CAM) (Zhou et al. 2016) uses Global Average Pooling (GAP) to identify important regions by scaling GAP weights. Grad-CAM (Selvaraju et al. 2017a) maintains model flexibility by using gradients of target classes in the last convolutional layer. Guided-GradCAM (Selvaraju et al. 2017b) combines Grad-CAM with Guided-BP for finer feature importance, while Relevance-CAM (Lee et al. 2021) uses LRP-based scores for detailed importance visualization.

Information Theory Approaches (white-box): Information Bottleneck for Attribution (IBA) (Schulz et al. 2020), based on information bottleneck theory, quantifies input feature contributions to model decisions. InputIBA (Zhang et al. 2021) improves IBA by achieving pixel-level heatmaps but can produce less intuitive and blurred maps.

Perturbation-based Methods (black-box): LIME (Ribeiro, Singh, and Guestrin 2016) approximates complex models with simpler ones for local interpretation but lacks holistic applicability in areas like law and finance. HSIC-Attribution (Novello, Fel, and Vigouroux 2022) uses the Hilbert-Schmidt Independence Criterion (HSIC) in Reproducing Kernel Hilbert Spaces (RKHS) to improve interpretability and efficiency. SMDL-Attribution (Chen et al. 2024) employs a submodular subset selection method to enhance image attribution for better model interpretability.

Methodology

Problem Formulation

Let $X \in \mathbb{R}^n$ be a random variable representing the input (e.g., image pixels), and let $f : \mathbb{R}^n \to \mathbb{R}$ be the model. Define $Y = f(X)$ as the output random variable. For a given realization $\boldsymbol{x} \in \mathbb{R}^n$ of X, the model output is $\boldsymbol{y} = f(\boldsymbol{x})$. The attribution map $A : \mathbb{R}^n \to \mathbb{R}^n$ assigns a relevance score $A_i(\boldsymbol{x})$ to each input feature $\boldsymbol{x}_i$, indicating the contribution of $\boldsymbol{x}_i$ to the prediction $f(\boldsymbol{x})$.

Let Z be a random variable that represents a bottlenecked, noisy version of X. For a given realization $\boldsymbol{z}$ of Z, The goal is to identify which parts of $\boldsymbol{x}$ are important by evaluating how much relevant information $\boldsymbol{z}$ retains while ensuring that the predictive confidence of $f(\boldsymbol{z})$ is not weakened.

To address this, we employ the information bottleneck (Tishby, Pereira, and Bialek 2000) method. The core idea of the information bottleneck method is to achieve both **sufficiency** and **minimality** in the representation Z (Dubois et al. 2020):

Sufficiency: By maximizing the mutual information $I[Y; Z]$ between Y and Z, we ensure that Z retains all the key features relevant to Y. Specifically, the set of sufficient representations $\mathcal{S}$ is defined as:

$$\mathcal{S} := \arg\max_{Z} I[Y; Z] \tag{1}$$

Minimality: Simultaneously, by minimizing the mutual information $I[X; Z]$ between X and Z, we ensure that Z discards unnecessary features. The set of minimal sufficient representations $\mathcal{M}$ is defined as:

$$\mathcal{M} := \arg\min_{Z \in S} I[X; Z] \tag{2}$$

To achieve both sufficiency and minimality in the representation Z, the information bottleneck method employs the following Lagrangian optimization formulation (Dubois et al. 2020):

$$\mathcal{L}_{\text{IB}} := -I[Y; Z] + \beta I[X; Z], \tag{3}$$

where the term β is a positive Lagrange multiplier that controls the trade-off between the sufficiency and the minimality. For the first term of the loss function, as reducing the cross-entropy loss $\mathcal{L}_{CE}$ effectively enhances a lower bound on the mutual information $I[Y; Z]$ (Alemi et al. 2016; Zhang et al. 2021), thereby the loss function for sufficiency can be defined as:

$$\mathcal{L}_{\mathcal{S}} = \mathcal{L}_{CE}. \tag{4}$$

However, computing $I[X; Z]$ is intractable. Typically, such calculations assume that Z follows a Gaussian distribution and has low dimensionality. This assumption does not hold in the pixel space, which is typically high-dimensional and non-Gaussian (Zhang et al. 2021). In the following section, we will introduce how a diffusion model can be used to derive a simple loss function that substitutes for the loss function of mutual information, thereby simplifying the optimization process.

Diffusion for Simplifying the Optimization of Information Bottlenecks

The process of reducing $I(X; Z)$ typically involves adding Gaussian noise to weaken the direct relationship between X and Z, thereby decreasing their shared information. This can be modeled using a diffusion framework, where Gaussian noise is progressively added to the input. Mathematically, this is described by sampling from the data distribution $\boldsymbol{x} \sim p(X = \boldsymbol{x})$ and incrementally adding noise, expressed as:

$$\boldsymbol{z} = \sqrt{\sigma(\alpha)}\boldsymbol{x} + \sqrt{\sigma(-\alpha)}\boldsymbol{\epsilon}, \tag{5}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$, σ is the standard sigmoid function, and α represents the log SNR. The noise term $\boldsymbol{\epsilon}$ obscures direct information about $\boldsymbol{x}$ within $\boldsymbol{z}$, leading to a reduction in $I(X; Z)$.

We further examine the diffusion model described by the equation $\boldsymbol{z}_\gamma = \sqrt{\gamma}\boldsymbol{x} + \boldsymbol{\epsilon}$, where γ represents the SNR with $\alpha = \log \gamma$. This formulation allows us to connect classical information theory principles to diffusion models. We establish the equivalence of this model with Equation 5, with a proof provided in the Appendix.

Referencing the classic result by (Guo, Shamai, and Verdú 2005; Kong, Brekelmans, and Ver Steeg 2023), the relationship between the derivative of mutual information with respect to γ and the minimum mean squared error (MMSE) is captured by the following equation:

$$\frac{d}{d\gamma} I(X = \boldsymbol{x}; Z = \boldsymbol{z}_\gamma) = \frac{1}{2}\text{mmse}(\gamma). \tag{6}$$

The MMSE quantifies the minimum achievable estimation error for the original signal $\boldsymbol{x}$ in the presence of noise $\boldsymbol{z}_\gamma$. It is formally defined as (Kong, Brekelmans, and Ver Steeg 2023):

$$\text{mmse}(\gamma) \equiv \min_{\hat{\boldsymbol{x}}(\boldsymbol{z}_\gamma, \gamma)} \mathrm{E}_{p(\boldsymbol{z}_\gamma, \boldsymbol{x})} \left[\|\boldsymbol{x} - \hat{\boldsymbol{x}}(\boldsymbol{z}_\gamma, \gamma)\|^2 \right]. \tag{7}$$

Here, $\hat{\boldsymbol{x}}(\boldsymbol{z}_\gamma, \gamma)$ is an estimator function that aims to reconstruct $\boldsymbol{x}$ from the noisy observations $\boldsymbol{z}_\gamma$. The optimal denoising function, $\hat{\boldsymbol{x}}^*$, which minimizes the estimation error, can be computed using the conditional expectation (Kong, Brekelmans, and Ver Steeg 2023):

$$\hat{\boldsymbol{x}}^*(\boldsymbol{z}_\gamma, \gamma) \equiv \mathrm{E}_{\boldsymbol{x} \sim p(\boldsymbol{x} | \boldsymbol{z}_\gamma)}[\boldsymbol{x}]. \tag{8}$$

Lemma 1: The mutual information parameterized by the SNR, $I(\boldsymbol{x}; \boldsymbol{z} \mid \gamma)$, can be expressed as:

$$I(\boldsymbol{x}; \boldsymbol{z} \mid \gamma) = \frac{1}{2} \int_0^\gamma \text{mmse}(\gamma')\, d\gamma'. \tag{9}$$

Proof. Please see the Appendix for the proof.

Lemma 2: Reducing γ leads to a decrease in mutual information.

Proof. Please see the Appendix for the proof.

To capture the varying importance of input features, we extend the scalar noise level γ to a tensor $\boldsymbol{\gamma}$ with the same dimensions as $\boldsymbol{x}$. This allows different noise levels $\boldsymbol{\gamma}_i$ to be assigned to each feature $\boldsymbol{x}_i$. The perturbed input is then defined as $\boldsymbol{z}_{\boldsymbol{\gamma}} = \sqrt{\boldsymbol{\gamma}} \odot \boldsymbol{x} + \boldsymbol{\epsilon}$, and the MMSE equation (7) is generally applicable to $\boldsymbol{\gamma}$ as a tensor. By optimizing $\boldsymbol{\gamma}$, we

Algorithm 1: Training $\boldsymbol{\alpha}_\theta$ for optimizing $\boldsymbol{z}$

An initialized U-Net-like model $\boldsymbol{\alpha}_\theta$
A classifier f with pre-trained parameters
An image $\boldsymbol{x}$ and its corresponding label $\boldsymbol{y}$
repeat
    Compute $\boldsymbol{\alpha} = \boldsymbol{\alpha}_\theta(\boldsymbol{x})$ ▷ U-Net predicts perturbation scale
    $\boldsymbol{z} = \sqrt{\sigma(\boldsymbol{\alpha})} \odot \boldsymbol{x} + \sqrt{\sigma(-\boldsymbol{\alpha})} \odot \boldsymbol{\epsilon}$ ▷ Generate a perturbed image
    $\hat{\boldsymbol{y}} = f(\boldsymbol{z})$ ▷ Classifier makes a prediction on perturbed input
    $\mathcal{L}_{CE} = -\sum \boldsymbol{y} \log \hat{\boldsymbol{y}}$ ▷ Cross-entropy loss for classifier output
    $\mathcal{L} = \mathcal{L}_{CE} + \beta \,\mathrm{mean}(\boldsymbol{\alpha})$ ▷ Total loss includes U-Net output
    Take gradient descent step on $\nabla_\theta \mathcal{L}$ ▷ Update U-Net parameters
until converged
return $\boldsymbol{\alpha}_\theta$ ▷ Return the trained U-Net model

can adaptively allocate noise levels to reflect feature importance: critical regions receive lower noise to retain information, while less relevant regions tolerate higher noise. And then we have the attribution map as follows,

$$A(\boldsymbol{x})_i = \frac{1}{2} \int_0^{\boldsymbol{\gamma}_i} \mathrm{mmse}(\boldsymbol{\gamma}')_i \, d\boldsymbol{\gamma}'_i. \tag{10}$$

Through rigorous proofs, we show that minimizing the SNR, $\boldsymbol{\gamma}$, is equivalent to minimizing mutual information $I(\boldsymbol{x}; \boldsymbol{z})$ in the optimization process. This approach directly addresses the minimality problem in the information bottleneck framework by simplifying the computational process and ensuring that the representation $\boldsymbol{z}$ retains only essential information from $\boldsymbol{x}$.

Loss function for minimality: To effectively implement this solution, we propose a new loss function that directly targets the minimization of $\boldsymbol{\gamma}$ as a surrogate for reducing $I(\boldsymbol{x}; \boldsymbol{z})$. This loss function is defined as:

$$\mathcal{L}_{\mathcal{M}} = \mathrm{mean}(\log \boldsymbol{\gamma}) = \mathrm{mean}(\boldsymbol{\alpha}). \tag{11}$$

The new loss function leads to the following optimization problem for the information bottleneck framework:

$$\mathcal{L}_{\mathrm{IB}} = \mathcal{L}_{\mathcal{S}} + \beta \mathcal{L}_{\mathcal{M}}. \tag{12}$$

Algorithm: Training $\boldsymbol{\alpha}_\theta$ for Optimizing the Representation $\boldsymbol{z}$ The training process for optimizing the intermediate representation $\boldsymbol{z}$ using the U-Net-like model $\boldsymbol{\alpha}_\theta$ is detailed as follows. The steps involve predicting a perturbation scale, generating a perturbed input, and updating the model parameters to minimize a combined loss function. For a complete overview of the procedure, refer to Algorithm 1.

MMSE Neural Estimator for Calculating $A(\boldsymbol{x})$

We now need to calculate $A(\boldsymbol{x})$ under the known conditions of $\boldsymbol{\alpha}$ after we train the $\boldsymbol{\alpha}_\theta$. This calculation relies on the derived integral formula (10), with a critical step being the

Algorithm 2: Training the MMSE denoiser $\boldsymbol{\epsilon}_\theta$

An initialized U-Net-like model $\boldsymbol{\epsilon}_\theta$
An image $\boldsymbol{x}$
repeat
    $\boldsymbol{\alpha} \sim \mathcal{N}(0, \boldsymbol{I})$
    $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$
    Take gradient descent step on $\nabla_\theta \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\sigma(\boldsymbol{\alpha})} \odot \boldsymbol{x} + \sqrt{\sigma(-\boldsymbol{\alpha})} \odot \boldsymbol{\epsilon}, \boldsymbol{\alpha})\|^2$
until converged
return $\boldsymbol{\epsilon}_\theta$ ▷ Return the trained model

assessment of the performance of the MMSE estimator at each noise level $\boldsymbol{\alpha}$ for optimally predicting the noise $\boldsymbol{\epsilon}$.

The goal of the MMSE estimator is to minimize the squared error between the noise $\boldsymbol{\epsilon}$ and its estimate $\hat{\boldsymbol{\epsilon}}(\boldsymbol{x}_{\boldsymbol{\alpha}})$, also known as the denoiser in diffusion models (Luo 2022). The denoiser is defined as (Kong et al. 2024):

$$\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\alpha}}(\boldsymbol{x}) \equiv \arg\min_{\hat{\boldsymbol{\epsilon}}(\cdot)} E_{p(\boldsymbol{x}),p(\boldsymbol{\epsilon})} \left[\|\boldsymbol{\epsilon} - \bar{\boldsymbol{\epsilon}}(\boldsymbol{x}_{\boldsymbol{\alpha}})\|^2 \right], \tag{13}$$

aiming to find a function $\hat{\boldsymbol{\epsilon}}$ that minimizes the sum of squared errors at all noise levels $\boldsymbol{\alpha}$. Although the prediction targets the noise $\boldsymbol{\epsilon}$, this process can equivalently be applied to predict the original signal $\boldsymbol{x}$, as the ability to accurately estimate $\boldsymbol{x}$ from the noise directly reflects the denoising model's performance.

Unlike traditional diffusion models like DDPM (Ho, Jain, and Abbeel 2020), where the noise level is predefined during training, we design $\boldsymbol{\alpha}$ as a tensor of the same dimension as the input $\boldsymbol{x}$, allowing it to represent arbitrary noise levels that vary across different pixels. A U-Net (Ronneberger, Fischer, and Brox 2015) architecture neural network $\boldsymbol{\epsilon}_\theta$, similar to that used in DDPM, is employed to implement $\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\alpha}}(\boldsymbol{x})$. Algorithm 2 displays the process to train the optimal denoiser.

The process of calculating mutual information involves first computing the MMSE across various noise levels $\boldsymbol{\alpha}$ through the denoiser neural network, by evaluating the network's performance across a range of $\boldsymbol{\alpha}$ values. Subsequently, the attribution map is calculated using the integral formula:

$$A(\boldsymbol{x})_i = \frac{1}{2} \int_0^{\boldsymbol{\alpha}_i} \left[\|\boldsymbol{\epsilon}_i - \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_{\boldsymbol{\alpha}'}, \boldsymbol{\alpha}')_i\|^2 \right] d\boldsymbol{\alpha}'_i, \tag{14}$$

involving the integration of MMSE from 0 to $\boldsymbol{\alpha}_i$. This computation requires precise estimation or numerical methods to perform.

To calculate the integral $A(\boldsymbol{x})_i$, the trapezoidal rule is then applied, which approximates the integral by linearly interpolating between points and summing the areas of the resulting trapezoids.

Experiments

In the experimental section, we perform a comprehensive comparison of our approach against several benchmark methods. First, we provide an intuitive comparison based on the generated attribution maps. Then, we evaluate the performance of each method across multiple quantitative metrics.

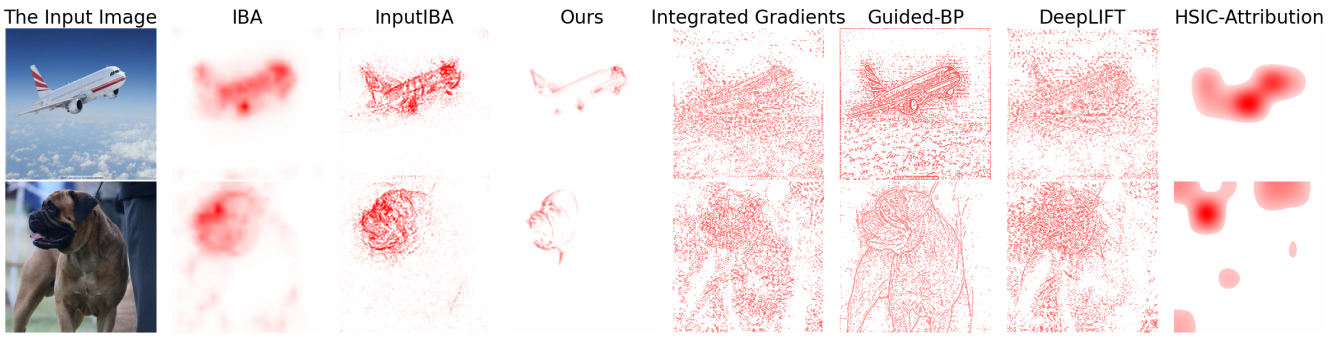

Figure 2: The visualization highlights various attribution methods and their impact on clarity and detail. IBA and HSIC-Attribution produce blurry images with imprecise contours, while DeepSHAP and Integrated Gradients offer fine-grained attributions but with chaotic, unfocused maps. InputIBA slightly improves edge definition but still allocates scores to background areas. In contrast, Guided-Backprop reveal sharp details but may misplace importance on background elements, potentially misleading interpretation. Our method stands out by providing clear, detailed maps that precisely outline the contours of predicted objects, enhancing interpretive accuracy.

Experimental Setup

For benchmarking, we selected IBA, InputIBA, Integrated Gradients, Guided-Backprop, DeepLIFT, and HSIC-Attribution. We used the original parameter settings provided by the authors in their respective publications. Specifically, for Guided-Backprop, Integrated Gradients, and DeepLIFT, we utilized the widely recognized Captum (Kokhlikyan et al. 2020) toolkit in PyTorch (Paszke et al. 2019). For Integrated Gradients, the `n_steps` parameter was set to 200. In the IBA and InputIBA methods, the β_{feat} parameter was set to 10. Additionally, for handling the second bottleneck parameter Λ in InputIBA, we set β_{input} to 20 and performed 60 iterations. For HSIC-Attribution, the grid size was set to 7.

For the models requiring explanation, we selected VGG16 (Simonyan and Zisserman 2014), Resnet50 (He et al. 2016), and EfficientNet (Tan and Le 2019), all with pretrained weights from the Torchvision (Marcel and Rodriguez 2010) package. We will present the results for VGG16 in the following sections, while the experimental results for the other models are provided in the Appendix.

Qualitative Comparision

The visualizations as shown in Figure 2 indicate that the attribution maps generated by IBA and HSIC-Attribution are relatively blurry and lack clarity in details. This is due to the interpolation into the input space, which results in a loss of precise contours and details of objects. While DeepSHAP and Integrated Gradients can generate fine-grained attributions, which are helpful for detailed analysis of model behavior, the resulting attribution maps appear chaotic and lack a clear focus. Although InputIBA improves upon the IBA method, the edges of the predicted objects in the images remain somewhat blurry, and some importance scores are still allocated to background areas.

Guided-Backprop, on the other hand, are able to reveal sharp edges and complex details within images, enhancing the visual impact of the attribution maps. However, there is a risk of misallocating importance scores to background elements rather than the primary objects of prediction, which can lead to misunderstandings about model behavior.

Our method produces clear and detailed attribution maps that accurately delineate the contours and details of predicted objects.

The Minimum Necessary Information Metric (MNIM)

To assess whether explainable methods meet the criteria of sufficiency and minimality, we utilize the Minimum Necessary Information (MNI (Fischer 2020)) criterion as a metric to compare explainable methods. This metric aims to highlight only the most critical parts identified by the explainable method, while the rest of the image is replaced by noise. The experimental setup is as follows:

1. Rank the pixel channel values in the attribution map according to their importance scores, from highest to lowest. Sequentially replace the corresponding pixel channel values in a fully noise-filled image with those from the original image based on this ranking.
2. During the replacement process, monitor the changes in model performance. Identify the number of pixel channel values N needed to achieve a threshold performance level (e.g., 90% prediction confidence). This N represents the minimum amount of information required to maintain the desired performance.
3. Divide N by the total number of pixel channel values in the image to obtain a ratio.

Figure 3 illustrates the number of pixels required by different methods to achieve a 90% confidence level in the model prediction. Our method requires only a minimal number of pixels to reach this confidence level, demonstrating that it effectively captures the most critical pixels.

We randomly selected 1,000 images from ImageNet (Deng et al. 2009) and conducted the aforementioned experiments. The statistical results are shown in Table 1. As seen

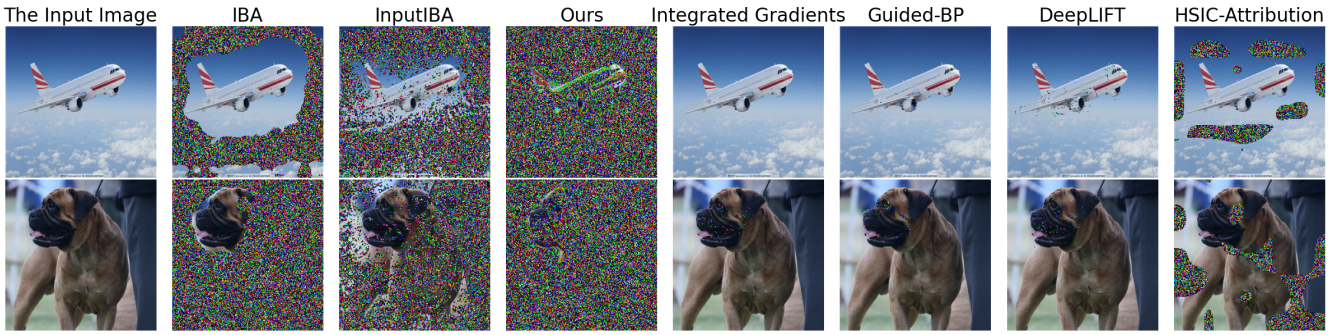

Figure 3: In qualitative comparisons of images with a 90% prediction confidence level, our method demonstrates the ability to achieve the set model prediction confidence with only minimal display of the predicted object's details.

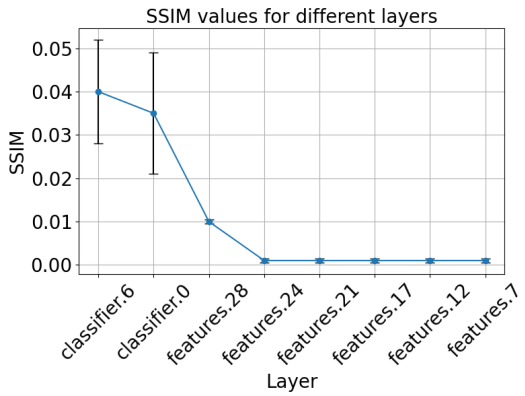

Figure 4: In each experiment, we randomly randomize the parameters of specified layers in the VGG16 model, starting from the back and moving forwards. The experimental results show that regardless of which layer we start the randomization from, the SSIM values remain consistently low.

in the table, our method achieves the model's performance with the full image using only about 2-3% of the pixel channel values, significantly outperforming other methods.

Parameter Randomization Sanity Check (Adebayo et al. 2018)

The purpose of the Parameter Randomization Sanity Check is to validate whether the attribution methods accurately and effectively explain model behavior, and the analysis is conducted using the Structural Similarity Index Metric (SSIM (Wang et al. 2004)). If the randomization of model parameters results in a lower SSIM value between the attribution map and the original, this suggests sensitivity to parameter changes, indicating that the attribution method effectively captures essential features of the model's decision-making process. The experimental results of our method presented in Figure 4 demonstrate that, regardless of which layer of the model we begin parameter randomization, the SSIM values are consistently low.

Method	MINM
IBA	0.165 ± 0.004
InputIBA	0.132 ± 0.002
Ours	$\mathbf{0.025} \pm 0.005$
Integrated Gradients	0.995 ± 0.002
Guided-BP	0.992 ± 0.005
Deep-Lift	0.995 ± 0.001
HSIC-Attribution	0.821 ± 0.007

Table 1: Our method can maintain the model's performance on the full image while using only about 2-3% of the pixel channel values, markedly surpassing other approaches. This highlights our method's ability to capture the most essential information in the image effectively.

Insertion and Deletion AUCs (Zhang et al. 2021)

The Deletion method starts with the original input and progressively removes pixels by replacing them with noise based on their importance scores. Conversely, the Insertion method gradually adds the most important pixels into a baseline image.

We assess the impact of these modifications by monitoring the model's predictions at each stage and calculating the Area Under the Curve (AUC) of the output across all steps. A larger difference between the Insertion AUC and the Deletion AUC indicates better performance of the attribution method. The Differences in AUCs (DAUCs) for various attribution methods are presented in Table 2. The results show that our method and InputIBA are the top performers, with IBA closely following. Other baseline methods perform significantly worse on this specific metric.

Bounding Boxes

We leverage human-annotated bounding boxes from the ImageNet dataset to quantify how well attribution methods identify and localize objects of interest. Specifically, we introduce a metric based on the importance scores generated by attribution methods, which measures the proportion of significant pixel channel values located within the bounding boxes.

First, we generate attribution maps and determine the

Method	DAUCs
IBA	0.774 ± 0.005
InputIBA	0.841 ± 0.002
Ours	$\mathbf{0.854} \pm 0.005$
Integrated Gradients	0.156 ± 0.003
Guided-BP	0.152 ± 0.008
Deep-Lift	0.163 ± 0.009
HSIC-Attribution	0.145 ± 0.002

Table 2: The results indicate that our method and InputIBA emerge as the best performers, followed closely by IBA. Notably, other baseline methods perform markedly worse on this specific metric.

minimum information N as defined by the MINM metric, which represents the smallest number of pixel channel values needed to maintain model performance. Next, we calculate how many of these N most important pixel channel values fall within the human-annotated bounding boxes, which typically enclose the primary object in the image. Finally, we compute the Box-Ratio, defined as the proportion of important pixel channel values within the bounding boxes $\frac{n}{N}$.

Method	Box-Ratio
IBA	0.997 ± 0.001
InputIBA	0.998 ± 0.001
Ours	0.998 ± 0.000
Integrated Gradients	0.691 ± 0.006
Guided-BP	0.698 ± 0.002
Deep-Lift	0.695 ± 0.005
HSIC-Attribution	0.903 ± 0.002

Table 3: The results indicate that our method and InputIBA emerge as the best performers, followed closely by IBA. Notably, other baseline methods perform markedly worse on this specific metric.

Table 3 demonstrates that IBA, InputIBA, and our method outperform other techniques. However, distinguishing between these three methods in terms of their superiority is challenging. The bounding boxes (bboxes) annotated by humans tend to cover large areas, including not only the object predicted by the model but also a significant surrounding space. Consequently, the most important pixel channel values identified by IBA, InputIBA, and our method predominantly fall within these expansive boundaries. To more accurately evaluate the effectiveness of explainable methods in identifying and localizing key information, we need to utilize more precise object boundaries than traditional bounding boxes.

Segmentation-based Ratio

We employ image segmentation masks to evaluate the effectiveness of attribution maps generated by explainability methods. Our evaluation utilizes semantic segmentation annotations from the FSS-1000 (Li et al. 2020) dataset, which provides detailed contours for objects within images. Compared to the rectangular frames of bounding boxes (BBOXs), semantic segmentation provides precise boundaries around visual objects, enabling a more accurate assessment of whether the most critical pixel channel values identified by explainability methods truly concentrate on features relevant to the model's decision-making. The experimental design and metric calculation are similar to the Box-Ratio metric, with the only difference being that bounding boxes are replaced by semantic segmentation regions.

Experimental results, as shown in Table 4, demonstrate that our method exhibits a leading performance, followed closely by IBA and InputIBA. These three methods employ information bottleneck techniques, effectively filtering out non-essential elements while retaining important pixel channel values. In contrast, other methods that do not utilize information bottleneck techniques have their most important pixel channel values covering almost the entire image, resulting in significantly poorer performance.

Method	SR
IBA	0.488 ± 0.004
InputIBA	0.468 ± 0.003
Ours	$\mathbf{0.511} \pm 0.002$
Integrated Gradients	0.079 ± 0.006
Guided-BP	0.078 ± 0.009
Deep-Lift	0.080 ± 0.002
HSIC-Attribution	0.377 ± 0.005

Table 4: The results indicate that our method outperforms others, with IBA and InputIBA closely trailing. These three methods leverage information bottleneck techniques, which adeptly eliminate non-critical elements and preserve essential pixel channel values. In contrast, other four methods tend to highlight pixel channel values across nearly the entire image, leading to substantially inferior performance.

Conclusion

In conclusion, this study introduces a novel method for generating attribution maps using diffusion models within the information bottleneck framework, offering a significant advancement in explainable AI. By effectively leveraging Gaussian noise from diffusion models, we demonstrated that minimizing SNR is equivalent to reducing mutual information, thereby simplifying the complex computation of mutual information in high-dimensional data. Our method achieves greater clarity and accuracy in attributions than existing techniques, requiring significantly fewer pixel values to maintain predictive confidence. The experimental results consistently highlighted the superior performance of our approach across various metrics, including the MINM, Insertion and Deletion AUCs, and segmentation-based evaluations. This work underscores the potential of diffusion models in enhancing the precision and explainability of AI-driven decisions, particularly in critical applications like healthcare, finance, and law, where explainability is paramount.

Acknowledgments

We acknowledge the Australian Government Research Training Program Scholarship. This work has been supported under project p2-47s by the SmartSat CRC, whose activities are funded by the Australian Government's CRC Program. This work has received partial support from the Australian Research Council Discovery Project (DP230101122).

References

Adebayo, J.; Gilmer, J.; Muelly, M.; Goodfellow, I.; Hardt, M.; and Kim, B. 2018. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31.

Alemi, A. A.; Fischer, I.; Dillon, J. V.; and Murphy, K. 2016. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*.

Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.-R.; and Samek, W. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7): e0130140.

Chaddad, A.; Peng, J.; Xu, J.; and Bouridane, A. 2023. Survey of explainable AI techniques in healthcare. *Sensors*, 23(2): 634.

Chen, R.; Zhang, H.; Liang, S.; Li, J.; and Cao, X. 2024. Less is More: Fewer Interpretable Region via Submodular Subset Selection. In *The Twelfth International Conference on Learning Representations*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Dubois, Y.; Kiela, D.; Schwab, D. J.; and Vedantam, R. 2020. Learning optimal representations with the decodable information bottleneck. *Advances in Neural Information Processing Systems*, 33: 18674–18690.

Fischer, I. 2020. The conditional entropy bottleneck. *Entropy*, 22(9): 999.

Gless, S. 2019. AI in the Courtroom: a comparative analysis of machine evidence in criminal trials. *Geo. J. Int'l L.*, 51: 195.

Guo, D.; Shamai, S.; and Verdú, S. 2005. Mutual information and minimum mean-square error in Gaussian channels. *IEEE transactions on information theory*, 51(4): 1261–1282.

Hanin, B. 2018. Which neural net architectures give rise to exploding and vanishing gradients? *Advances in neural information processing systems*, 31.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

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

Kokhlikyan, N.; Miglani, V.; Martin, M.; Wang, E.; Alsallakh, B.; Reynolds, J.; Melnikov, A.; Kliushkina, N.; Araya, C.; Yan, S.; and Reblitz-Richardson, O. 2020. Captum: A unified and generic model interpretability library for PyTorch. arXiv:2009.07896.

Kong, X.; Brekelmans, R.; and Ver Steeg, G. 2023. Information-Theoretic Diffusion. In *International Conference on Learning Representations*.

Kong, X.; Liu, O.; Li, H.; Yogatama, D.; and Steeg, G. V. 2024. Interpretable Diffusion via Information Decomposition. In *The Twelfth International Conference on Learning Representations*.

Lee, J. R.; Kim, S.; Park, I.; Eo, T.; and Hwang, D. 2021. Relevance-cam: Your model already knows where to look. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14944–14953.

Li, X.; Wei, T.; Chen, Y. P.; Tai, Y.-W.; and Tang, C.-K. 2020. Fss-1000: A 1000-class dataset for few-shot segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2869–2878.

Luo, C. 2022. Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970*.

Marcel, S.; and Rodriguez, Y. 2010. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM international conference on Multimedia*, 1485–1488.

Novello, P.; Fel, T.; and Vigouroux, D. 2022. Making sense of dependence: Efficient black-box explanations using dependence measure. *Advances in Neural Information Processing Systems*, 35: 4344–4357.

Pan, D.; Li, X.; and Zhu, D. 2021. Explaining deep neural network models with adversarial gradient integration. In *Thirtieth International Joint Conference on Artificial Intelligence (IJCAI)*.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, 8024–8035. Curran Associates, Inc.

Petsiuk, V.; Das, A.; and Saenko, K. 2018. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. " Why should i trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1135–1144.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, 234–241. Springer.

Saraswat, D.; Bhattacharya, P.; Verma, A.; Prasad, V. K.; Tanwar, S.; Sharma, G.; Bokoro, P. N.; and Sharma, R. 2022. Explainable AI for healthcare 5.0: opportunities and challenges. *IEEE Access*, 10: 84486–84517.

Schulz, K.; Sixt, L.; Tombari, F.; and Landgraf, T. 2020. Restricting the Flow: Information Bottlenecks for Attribution. In *International Conference on Learning Representations*.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017a. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, 618–626.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017b. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, 618–626.

Shrikumar, A.; Greenside, P.; and Kundaje, A. 2017. Learning important features through propagating activation differences. In *International conference on machine learning*, 3145–3153. PMLR.

Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2014. Visualising image classification models and saliency maps. *Deep Inside Convolutional Networks*, 2.

Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Smilkov, D.; Thorat, N.; Kim, B.; Viégas, F.; and Wattenberg, M. 2017. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*.

Soundararajan, R.; and Shenbagaraman, V. 2024. Enhancing Financial Decision-Making Through Explainable AI And Blockchain Integration: Improving Transparency And Trust In Predictive Models. *Educational Administration: Theory and Practice*, 30(4): 9341–9351.

Springenberg, J. T.; Dosovitskiy, A.; Brox, T.; and Riedmiller, M. 2014. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*, 3319–3328. PMLR.

Tan, M.; and Le, Q. 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, 6105–6114. PMLR.

Tishby, N.; Pereira, F. C.; and Bialek, W. 2000. The information bottleneck method. *arXiv preprint physics/0004057*.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4): 600–612.

Yang, Q.; Zhu, X.; Fwu, J.-K.; Ye, Y.; You, G.; and Zhu, Y. 2021. Mfpp: Morphological fragmental perturbation pyramid for black-box model explanations. In *2020 25th International conference on pattern recognition (ICPR)*, 1376–1383. IEEE.

Zhang, Y.; Khakzar, A.; Li, Y.; Farshad, A.; Kim, S. T.; and Navab, N. 2021. Fine-grained neural network explanation by identifying input features with predictive information. *Advances in Neural Information Processing Systems*, 34: 20040–20051.

Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; and Torralba, A. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2921–2929.