Journal of King Saud University Computer and Information Sciences (2026) 38:126
https://doi.org/10.1007/s44443-026-00507-x

RESEARCH

MedExplainer: an interpretable ensemble parallel-tree framework for interpreting vision–language models in medical imaging

Pei Xu[1] · Chung-You Tsai[2,3] · Chih-Yung Chang[4] · Yu-Ting Chih[5] · Diptendu Sinha Roy[6]

Received: 18 November 2025 / Accepted: 21 January 2026

Abstract

Large multimodal vision–language models (VLMs) have achieved remarkable success in image and video analysis. However, their inherent black-box nature limits their applicability in medical imaging, where interpretability is critical. Although several existing explainable machine learning techniques partially address this issue, they often suffer from model dependency, interpretability randomness, and high computational complexity. To overcome these challenges, this paper proposes MedExplainer, an innovative explainable ensemble parallel-tree framework. The method first generates a set of perturbed samples by applying fixed masking operations to individual images or video frames, and then obtains prediction scores through the VLM. Using Parallel bagging and boosting strategies, MedExplainer constructs parallel decision trees and visualizes feature importance for local interpretability, while aggregating all sample-level explanations to produce a comprehensive global interpretation. The experiment evaluated the method on multiple medical image and video datasets, and the results demonstrate that the proposed MedExplainer provides stronger and more consistent explanations than existing methods. Interestingly, this design also enables the enhancement of lightweight VLMs for medical image segmentation tasks. In particular, the MedExplainer-augmented medicalGemma-4B model outperforms Google's recently released Gemini Nano Banana on segmentation benchmarks. Demonstration videos are available in the supplementary materials.

Keywords Explain · Interpretable machine learning · Medical-image · Parallel decision trees

1 Introduction

Large Vision-Language Models (VLMs) have shown unprecedented performance in processing unstructured medical data (Chin et al. 2025; Jiang et al. 2025). Consequently, integrating them into healthcare has shifted from theoretical exploration to practical necessity. However, a critical "trust bottleneck" currently hinders their clinical application. Modern VLMs have largely solved the problem of predictive accuracy, often matching or exceeding human experts. Yet, they fail to satisfy the requirement for explanatory fidelity (Zeng et al. 2025).

It is vital to distinguish between "decision correctness" and "explanation fidelity." This distinction is often overlooked. As shown in Fig. 1(a), a VLM (e.g., Gemini 2.5 Pro) correctly diagnoses a bladder lesion from an endoscopic image. However, as shown in Fig. 1(b), current interpretabil-

✉ Chih-Yung Chang
cychang@mail.tku.edu.tw

Pei Xu
xupei@hfuu.edu.cn

Chung-You Tsai
pgtsai@gmail.com

Yu-Ting Chih
143221@o365.tku.edu.tw

Diptendu Sinha Roy
diptendu.sr@nitm.ac.in

[1] the School of Artificial Intelligence and Big Data, Hefei University, Hefei, China

[2] Far Eastern Memorial Hospital, New Taipei, Taiwan

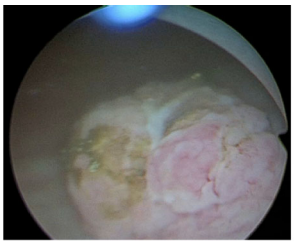
[3] Department of Electrical Engineering, Yuan Ze University, Taoyuan, Taiwan

[4] Department of Computer Science and Information Engineering, Tamkang University, New Taipei, Taiwan

[5] Department of Artificial Intelligence, Tamkang University, New Taipei, Taiwan

[6] Department of Computer Science and Engineering, National Institute of Technology Meghalaya, Shillong, India

Published online: 11 February 2026

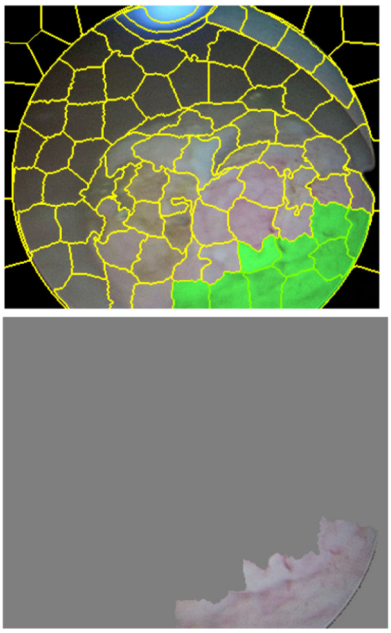


Q: What is the illness depicted in the picture?

A: Based on the visual evidence, the image appears to be from a medical procedure called a cystoscopy, which is an endoscopic examination of the bladder.

The image shows a large, lumpy, and fleshy growth on the inner lining of what is likely the bladder. This type of growth is highly characteristic of a bladder tumor.

(a)



(b)

Fig. 1 Example of LLM interpretation failure. (**a**) Correct identification. (**b**) Interpretable visualization showing focus on the wrong region

ity tools highlight a nearby normal protrusion rather than the tumor itself. This does not necessarily mean the model relied on irrelevant features to make the prediction. Instead, it indicates that the interpretation method failed to faithfully visualize the true decision boundary. This "fidelity gap" poses a severe risk in medicine. Without clinical validation, "black-box" correctness is insufficient (Jiang et al. 2024).

Existing literature attempts to bridge this gap but faces significant methodological limitations, particularly regarding stability. Figure 2 demonstrates the issue of Interpretability Stochasticity in current perturbation-based methods.

We analyze the same input image five times using a standard explanation algorithm. As shown in Fig. 2, the resulting visualizations (green highlighted regions) fluctuated drastically over runs. The focus shifted from the lesion to unrelated tissue without a consistent pattern. This instability implies that for the same patient x, the explanation result $Explanation(x)_i$ differs from $Explanation(x)_j$. This severely undermines clinical reproducibility. Physicians cannot rely on diagnostic aids that change their evidence based on random seeds (Lee et al. 2025). Furthermore, many existing tools are tightly coupled with specific CNN architectures (Selvaraju et al. 2020). Others suffer from excessive computational costs due to high-dimensional gradient calculations (Lundberg and Lee 2017).

To address these challenges—specifically the fidelity gap illustrated in Fig. 1 and the explanation instability shown in Fig. 2—this paper introduces **MedExplainer**, a model-agnostic interpretability framework for medical vision–language models. The central methodological contribution of this work lies in replacing stochastic sampling schemes with a deterministic ensemble-based mechanism,enabling empirically more stable and reproducible explanations under fixed perturbation and segmentation settings, while remaining architecture-independent. Concretely, the proposed MedExplainer is built upon the following key design principles.

- **Deterministic Masking Strategy.** We propose a deterministic masking strategy based on a fixed perturbation ratio, which significantly reduces the randomness commonly observed in conventional perturbation-based methods by replacing stochastic sampling with a deterministic masking strategy. By decoupling explanation generation from the underlying model architecture, this strategy allows MedExplainer to operate without requiring gradient access, thereby supporting a wide range of black-box vision–language models.
- **Ensemble Design Inspired by Bagging and Boosting Principles.** The proposed ensemble design is motivated by variance reduction and hard-region emphasis principles commonly associated with bagging and boosting, while adopting an independent and parallel training scheme to improve computational efficiency.
- **Parallelized Optimization Scheme.** From an algorithmic perspective, we design a parallelized optimization scheme that restructures the perturbation evaluation process into independent, concurrently executable

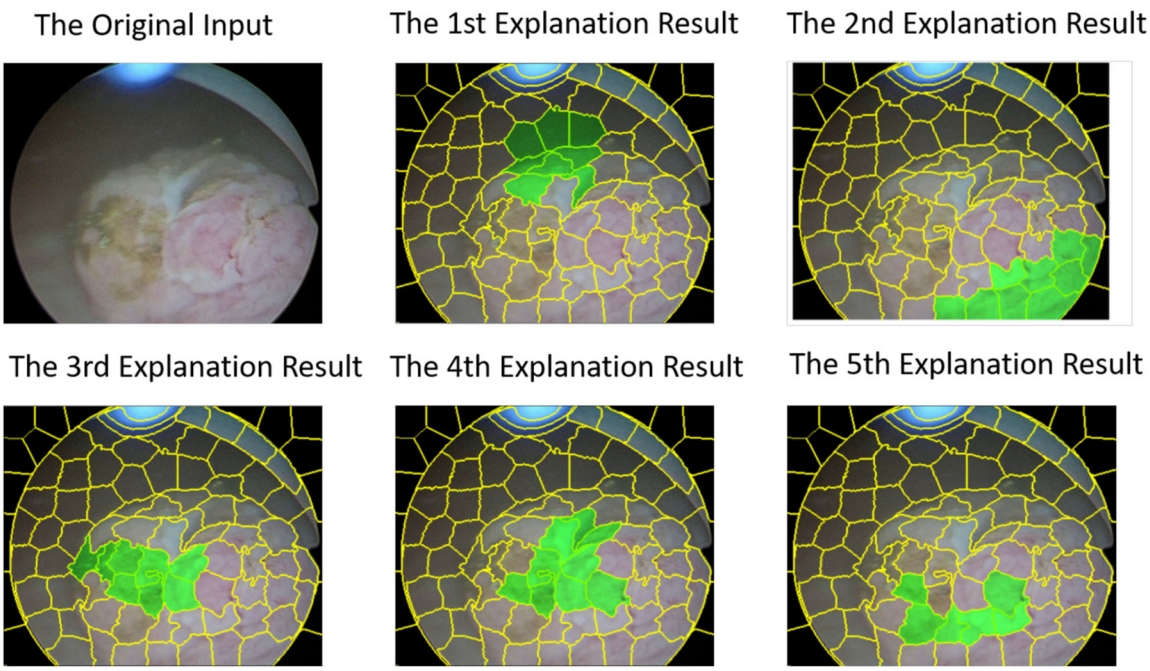


Fig. 2 The problem of randomness in explanation

components. This design is motivated by the goal of reducing latency bottlenecks commonly associated with sequential tree-based surrogates, by enabling fully parallel training and inference.

With respect to data usage and the evaluation protocol, the quantitative benchmarking of MedExplainer is conducted exclusively on standard medical imaging datasets. Specifically, all primary experimental results are obtained on the Radiology Objects in COntext (ROCO) dataset (Pelka et al. 2018), the UDIAT breast ultrasound dataset (Yap et al. 2017), and the COVID-19 CT Scans dataset (Rahimzadeh et al. 2021). These datasets constitute the foundation for all quantitative evaluations of explanation fidelity, stability, and computational efficiency reported in the main results section.

In contrast, medical surgical video data, including cystoscopic examination videos, are used solely for qualitative case studies. These video-based experiments are intended to illustrate the practical applicability and interpretability behavior of the proposed method in real-world clinical scenarios, rather than to serve as sources of quantitative performance metrics. Accordingly, all principal conclusions of this paper are grounded in visual and numerical evidence derived from the benchmark medical imaging datasets used for quantitative scoring, thereby ensuring a clear separation between controlled quantitative evaluation and illustrative qualitative case analysis.

The remainder of this paper is organized as follows. Section 2 reviews the limitations and gaps in the existing literature. Section 3 details the design and integration logic of the proposed MedExplainer framework. Section 4 presents quantitative evaluations focusing on explanation fidelity and stability. Section 5 discusses the experimental findings and outlines the limitations of the proposed approach. Finally, Section 6 concludes the paper and highlights directions for future work.

2 Related work

This section reviews the related work and is organized into three parts. The first part introduces perturbation-based explainable machine learning methods. The second part focuses on gradient-based visualization approaches. The third part discusses several applications of vision–language models (VLMs) in the field of medical imaging.

2.1 Perturbation-driven approaches

Perturbation-driven techniques explain model behavior by modifying specific parts of an input instance and observing how the prediction changes. By comparing these variations, the method infers which features play decisive roles in the decision-making process. For example, Zeiler and Fergus (2014) localized discriminative regions using a sliding occlusion window, while (Ribeiro et al. 2016) applied randomized masks on images or text to fit a local surrogate model. Petsiuk et al. (2018) further adopted stochastic masking to estimate feature relevance. Shapley-value–based attribution has also become a prominent direction: Ancona et al. (2019)

approximated feature contributions using cooperative-game–theoretic formulations, and Lundberg and Lee (2017) incorporated probabilistic Shapley derivatives to analyze decision behavior. Additional advances include clustering-based perturbations (Zafar and Khan 2021), tree-structured SHAP integration (Liu et al. 2024), and improved stability in G-LIME through an Elastic-Net estimator (Li et al. 2023). Furthermore, Tan et al. (2024) proposed an unbiased-sampling variant of GLIME to achieve higher local fidelity. More recent works also address structural limitations of surrogate models. For instance, Lee et al. (2025) developed EnEXP by replacing LIME's linear model with Random Forest and XGBoost proxies to mitigate multicollinearity, and Jiang et al. (2025)) introduced RealEXP, which reformulates Shapley attributions into self-importance and interaction-importance components within a LIME-like local explanation framework.

Despite their generality, perturbation-driven methods still face two fundamental limitations. **First, explanation instability remains a major issue.** As illustrated in Fig. 2, when explaining why an image is classified as bladder cancer, the model produces five noticeably different attribution maps across repeated perturbation runs. These inconsistencies reveal that perturbation-driven explanations are highly sensitive to randomness, often producing contradictory or irreproducible importance regions. When identical inputs yield diverging explanations, domain experts cannot reliably determine which features genuinely influence the model's decision, severely undermining interpretive trustworthiness.

Second, these methods incur extremely high computational costs. Shapley-value–based approaches require enumerating all subsets of features, which is computationally infeasible; for instance, with 60 attributes, full enumeration requires evaluating $60! \approx 8.32 \times 10^{81}$, far beyond practical computing capability. Although approximation strategies exist, they remain computationally intensive in high-dimensional settings. Tree-ensemble surrogate approaches, such as those based on Random Forests or XGBoost, must train large ensembles before generating explanations, substantially increasing inference latency. These computational burdens limit the applicability of perturbation-driven interpretability, particularly in real-time or resource-constrained environments.

2.2 Gradient-oriented visualization methods

The primary aim of these approaches is to identify which input features meaningfully influence a model's output by examining one or multiple backpropagation processes. Through gradient-based analyses, they seek to determine which components of the input data are most relevant to the model's predictions. For instance, Zhang et al. (2018) traced the activation-related input pixels for specific nodes or categories to elucidate how the network makes decisions, offering a direct view of how input features relate to the network structure. Selvaraju et al. (2017) introduced Grad-CAM, which produces visual heatmaps from convolutional layer gradients to highlight regions essential to the model's decisions. Binder et al. (2016) proposed distributing output errors proportionally back to input features to quantify their influence. Shrikumar et al. (2017) assigned "importance scores" by comparing each feature's contribution with that of a reference input. Smilkov et al. (2017) improved visualization clarity by injecting noise into the input and averaging the resulting gradients. Zahavy et al. (2016) used Jacobian Saliency Maps to emphasize input pixels that strongly affect value or policy predictions. Bakchy et al. (2024) extended gradient-based interpretability with a lightweight CNN that integrates Grad-CAM for faster and more efficient analysis. Niaz et al. (2024) proposed Increment-CAM, combining Grad-CAM, a modified Score-CAM, heatmap fusion, and regularization. Li et al. (2024) introduced CR-CAM, which generates class activation maps for CNNs and GCNs by comparing feature-map distances in manifold space to lessen the weight of surrounding regions.

Although these techniques provide intuitive and informative visual tools for understanding deep learning models, they still encounter several challenges:

Difficulty interpreting extremely large models. As model size grows, gradient signals often become diluted or noisy, complicating efforts to attribute importance to specific input features. In Transformer-based LLMs and VLMs, for example, gradients must traverse many stacked layers and multiple attention heads. This deep and parallel architecture can weaken gradient clarity, making it hard to capture long-range dependencies or determine how different heads contribute to the final prediction. The distributed nature of attention further complicates gradient-based explanations because diverse semantic patterns may be spread across heads, reducing the interpretability of any single gradient map.

Limited ability to deliver global explanations. Gradient-based approaches primarily offer local insights for individual samples. While effective for explaining isolated predictions, they fall short when describing broader patterns across entire datasets. They cannot easily reveal how a model behaves on different data segments or summarize the overarching structures it learns. This limitation is especially consequential in fields such as healthcare or finance, where understanding global model behavior is as important as explaining individual predictions.

Table 1 presents a comparison between the proposed method, MedExplainer, and previous approaches, analyzed across three dimensions: stability, low cost, and global

Table 1 Compare with the related works

Related Works	Type	Stability	Low Cost	Global Explanations
(Zeiler and Fergus 2014)	Perturb-Based	X	Y	X
(Ribeiro et al. 2016)	Perturb-Based	X	X	X
(Petsiuk et al. 2018)	Perturb-Based	X	X	X
(Ancona et al. 2019)	Perturb-Based	X	X	X
(Lundberg and Lee 2017)	Perturb-Based	Y	X	X
(Zafar and Khan 2021)	Perturb-Based	X	X	X
(Liu et al. 2024)	Perturb-Based	X	X	X
(Li et al. 2023)	Perturb-Based	X	X	X
(Tan et al. 2024)	Perturb-Based	X	X	X
(Lee et al. 2025)	Perturb-Based	X	X	≡
(Jiang et al. 2025)	Perturb-Based	X	X	≡
(Zhang et al. 2018)	Gradient-Oriented	Y	X	X
(Selvaraju et al. 2017)	Gradient-Oriented	Y	X	X
(Smilkov et al. 2017)	Gradient-Oriented	Y	X	X
(Binder et al. 2016)	Gradient-Oriented	Y	X	X
(Bakchy et al. 2024)	Gradient-Oriented	Y	X	X
(Niaz et al. 2024)	Gradient-Oriented	Y	X	X
(Li et al. 2024)	Gradient-Oriented	Y	X	X
The Proposed Method	Perturb-Based	Y	Y	Y

explanations. In the table, Y indicates the presence of a given property, X denotes the absence of the property, and ≡ represents partial effectiveness.

2.3 Vision Language Models (VLMs) in the field of medical imagings

Deep learning approaches have achieved remarkable performance across diverse modalities by leveraging complex feature extraction and alignment mechanisms. In the medical domain, Wang et al. (2024) proposed a meta-entity-driven triplet mining strategy to align medical vision–language models, significantly enhancing the semantic consistency between radiological images and reports. Addressing the challenges of medical imaging reconstruction, Liao et al. (2024) introduced SUFFICIENT, a scan-specific unsupervised framework capable of reconstructing high-resolution 3D isotropic fetal brain MRI without requiring extensive paired ground-truth data. Advancements have also extended to dynamic video understanding and natural language generation. For instance, Hu et al. (2023) developed a feature disentanglement approach for signer-independent sign language recognition, effectively separating signer-specific attributes from linguistic content to improve generalization. Similarly, in the field of image captioning, Ceylan et al. (2024) presented TRCaptionNet++, an optimized encoder–decoder architecture fine-tuned on large-scale datasets to generate high-quality Turkish captions.

However, despite the impressive accuracy and efficacy demonstrated by these state-of-the-art models, a critical limitation persists: the lack of interpretability. Most of these architectures operate as "black boxes," where the internal decision-making process remains opaque to human users. For example, while feature disentanglement and triplet mining improve metric performance, it is often unclear which specific visual regions or linguistic tokens contribute most to the final prediction. This opacity is particularly concerning in high-stakes fields such as medical diagnosis (Wang et al. 2024; Liao et al. 2024), where clinical practitioners require transparent reasoning to trust and validate algorithmic outputs. Consequently, the inability to explain why a model reaches a specific conclusion hinders the broader adoption and reliability of these advanced deep learning systems in real-world applications.

3 The proposed MedExplainer

This chapter provides a detailed introduction to the proposed MedExplainer. Unlike previous works (Zeiler and Fergus 2014; Ribeiro et al. 2016; Petsiuk et al. 2018; Lee et al. 2025; Tan et al. 2024; Jiang et al. 2025; Li et al. 2024; Bakchy et al. 2024; Niaz et al. 2024),

the proposed MedExplainer offers three major advantages: instability, low Cost, and global Explanations. As shown in Fig. 3, the proposed MedExplainer is composed of five modules: perturb samples and prediction generation, the

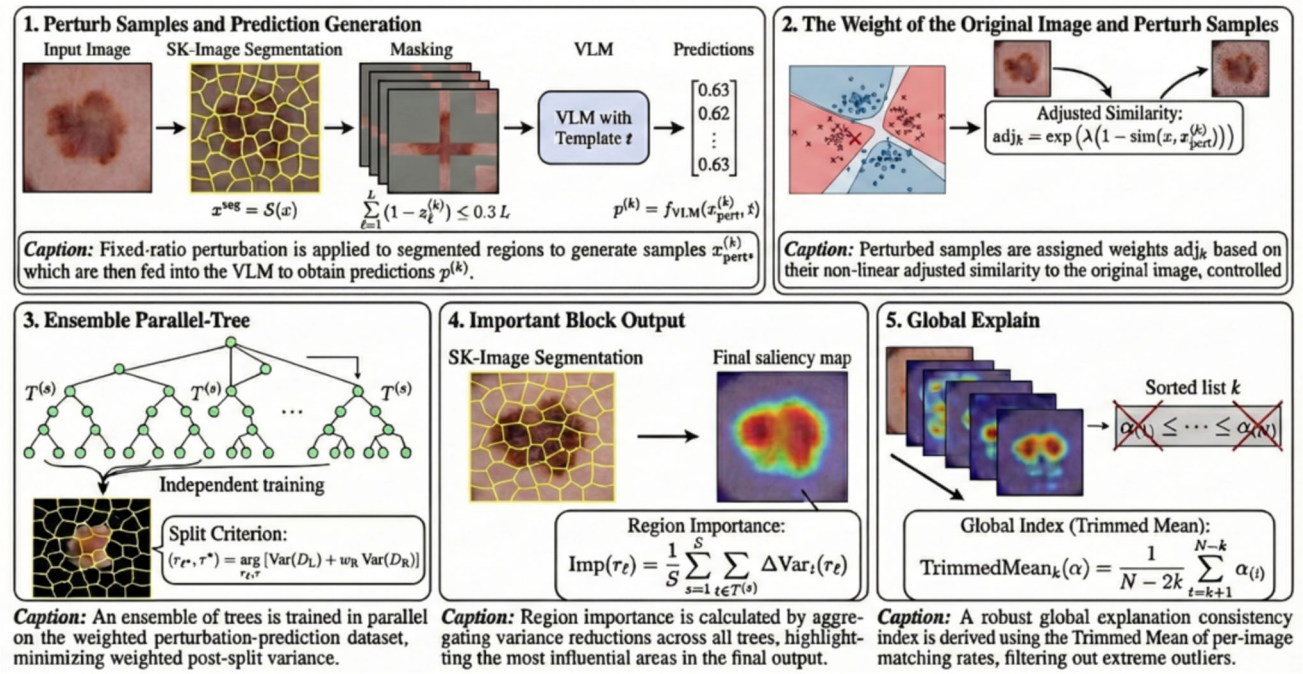


Fig. 3 The workflow of the proposed MedExplainer. The figure illustrates the complete pipeline of the proposed MedExplainer framework for explaining vision–language models (VLMs) in medical imaging. (1) Given an input medical image, superpixel-based segmentation is first applied, and fixed-ratio perturbations are generated by masking different segmented regions. These perturbed samples are then fed into the VLM to obtain corresponding prediction scores. (2) Each perturbed sample is assigned a weight based on its non-linearly adjusted similarity to the original image, ensuring that samples more similar to the original image contribute more significantly to the explanation. (3) An ensemble of parallel decision trees is trained on the weighted perturbation–prediction pairs, capturing diverse decision patterns while minimizing weighted post-split variance. (4) Region importance is computed by aggregating variance reductions across all trees in the ensemble, resulting in a final saliency map that highlights the most influential image regions for the model's prediction. (5) A global explanation consistency index is further derived using a trimmed mean strategy over per-image importance scores, providing a robust global explanation while reducing the influence of outliers

weight for the original image and perturb samples, ensemble Parallel-Tree, important block Output, and global explain. The details of each module are described in the following sections. Table 2 presents the key terms used in our method and their corresponding descriptions.

3.1 Perturb samples and prediction generation

This section introduces the *perturb samples and prediction generation* module. The purpose of this component is to systematically generate perturbed samples from the input image and examine how the model's predictions change when different regions are modified. This enables the assessment of each region's importance in the model's final decision. In other words, this step forms the foundation of regional importance analysis: if masking a specific region leads to a substantial shift in the model's output, that region is considered critical to the model's inference.

Unlike previous studies such as Ribeiro et al. (2016); Zafar and Khan (2021), which employ random masking or clustering-based perturbation methods, the proposed approach utilizes a *fixed-percentage masking* strategy. This ensures that each perturbation maintains a consistent degree of modification, thereby improving the stability and reproducibility of the subsequent modeling process.

In practice, the input image is first segmented into structured regions using SK-Image. Perturbed samples are then generated by progressively masking different numbers of regions according to predefined fixed ratios. The original image and all perturbed images are subsequently fed into the VLM model, where prediction probabilities are obtained through a designated template. This procedure produces a dataset composed of perturbed images along with their corresponding prediction scores.

Specifically, the *perturb samples and prediction generation* module consists of the following four steps: segmentation, fixed-ratio masking, prediction generation and perturbation database construction. The detailed procedures are described as follows.

The first step is segmentation. The aims of this step are to apply superpixel segmentation to obtain structured regions. Let x denote the input image in Fig. 3, and let $\mathcal{S}$ represent

Table 2 Notation summary

Symbol	Description
x	Input image.
$S(\cdot)$	Superpixel segmentation operator.
x_{seg}	Segmented representation of x: $x_{\text{seg}} = S(x)$.
$R = \{r_1, \ldots, r_L\}$	Set of L segmented regions (superpixels).
L	Number of regions/superpixels ($\lvert R \rvert$).
$z^{(k)}$	k-th binary mask over regions; $z_\ell^{(k)} \in \{0, 1\}$.
K	Number of perturbations / masks / samples.
ρ	Masking ratio (controls how many regions are hidden).
$M(\cdot, \cdot)$	Masking operator applied to segmented image.
$x_{\text{pert}}^{(k)}$	k-th perturbed image: $x_{\text{pert}}^{(k)} = M(x_{\text{seg}}, z^{(k)})$.
D_{Wiki}	Medical text-description database (curated from Wikipedia disease pages).
t	Selected description/prompt text, with $t \in D_{\text{Wiki}}$.
$f_{\text{VLM}}(\cdot, \cdot)$	Vision–language model mapping (image, text) to a probability.
$p^{(0)}$	VLM prediction on original sample: $p^{(0)} = f_{\text{VLM}}(x, t)$.
$p^{(k)}$	VLM prediction on perturbed sample: $p^{(k)} = f_{\text{VLM}}(x_{\text{pert}}^{(k)}, t)$.
sim_k	Similarity proxy: fraction of regions retained in mask $z^{(k)}$.
adj_k	Adjusted kernel weight used for perturbation k (e.g., exponential form).
λ	Kernel/weight hyperparameter controlling the decay strength.
X_k	Weighted feature vector for perturbation k (region-level features after weighting).
$\mathcal{T} = \{T^{(1)}, \ldots, T^{(S)}\}$	Ensemble of S parallel decision trees.
$T^{(s)}$	The s-th decision tree.
$\text{Imp}(r_\ell)$	Importance score for region r_ℓ aggregated from the tree ensemble.
R^2	Coefficient of determination between black-box outputs and surrogate predictions.
$\text{Adj}R^2$	Adjusted R^2 accounting for model complexity.
Stability	Stability score (averaged Jaccard overlap of selected important regions).
TrimmedMean_k	Trimmed mean after removing k smallest and k largest values.
Dice	Dice score used for segmentation overlap evaluation.

the SK-Image–based superpixel segmentation operator. The segmentation step generates a structured representation:

$$x^{\text{seg}} = \mathcal{S}(x), \tag{1}$$

which can be decomposed into a finite set of regions:

$$\mathcal{R} = \{r_1, r_2, \ldots, r_L\}, \tag{2}$$

where each region r_ℓ ($1 \le \ell \le L$) corresponds to one superpixel block depicted in the SK-Image panel of Fig. 3.

The second step is fixed-ratio masking. The purpose of this step is to feed both the original image and its perturbed versions into the VLM and extract the corresponding prediction probabilities using a predefined template. To construct the perturbed samples, we define a set of binary masks.

$$\mathcal{Z} = \{z^{(1)}, z^{(2)}, \ldots, z^{(K)}\}, \tag{3}$$

where each mask is given by

$$z^{(k)} = (z_1^{(k)}, z_2^{(k)}, \ldots, z_L^{(k)}), \qquad z_\ell^{(k)} \in \{0, 1\}. \tag{4}$$

Each binary indicator specifies whether the corresponding region is masked or preserved:

$$z_\ell^{(k)} = \begin{cases} 0, & \text{if region } r_\ell \text{ is masked,} \\ 1, & \text{if region } r_\ell \text{ is retained.} \end{cases} \tag{5}$$

To implement the fixed-percentage masking strategy, all masks satisfy:

$$\sum_{\ell=1}^{L} (1 - z_\ell^{(k)}) = \lfloor 0.3\, L \rfloor, \tag{6}$$

which enforces a fixed-percentage masking strategy by requiring that *exactly* 30% of the regions are masked in each perturbed image. Specifically, (6) sets the number of masked regions to $\lfloor 0.3L \rfloor$ (or an equivalent integer rounding rule), ensuring that every perturbation removes a constant 30% portion of the total regions rather than an upper-bounded or tunable amount.

Applying mask $z^{(k)}$ to the segmented image produces a perturbed sample:

$$x_{\text{pert}}^{(k)} = \mathcal{M}\left(x^{\text{seg}}, z^{(k)}\right). \tag{7}$$

The full set of perturbed images is thus:

$$\mathcal{X}_{\text{pert}} = \left\{x_{\text{pert}}^{(1)}, x_{\text{pert}}^{(2)}, \ldots, x_{\text{pert}}^{(K)}\right\}. \tag{8}$$

As shown in Fig. 3, the textual descriptions used for prompting the VLM are retrieved from a curated medical database constructed from Wikipedia disease articles. Let the selected description be denoted by:

$$t \in \mathcal{D}_{\text{Wiki}}, \tag{9}$$

where $\mathcal{D}_{\text{Wiki}}$ contains clinical symptom descriptions of tumors, polyps, warts, inflammations, and other related diseases.

A unified prompting template is adopted for all samples:

> *"You are now a classifier. Please tell me the probability that the image matches the given text. The probability is between 0 and 1, rounded to two decimal places."*

This template ensures consistent inference conditions for both original and perturbed samples.

The third step is prediction generation. Input both the original and perturbed images into the VLM and extract prediction probabilities via the template. Let f_{VLM} denote the vision–language model (VLM). The original image and each perturbed image are paired with the same text t under the same template, producing:

$$p^{(0)} = f_{\text{VLM}}(x, t), \tag{10}$$

$$p^{(k)} = f_{\text{VLM}}\left(x_{\text{pert}}^{(k)}, t\right), \qquad k = 1, \ldots, K. \tag{11}$$

The fourth step is perturbation database construction. Collecting all prediction values yields the perturbation–prediction dataset:

$$\mathbf{p} = (p^{(0)}, p^{(1)}, \ldots, p^{(K)}). \tag{12}$$

This dataset supports ensemble modeling and global interpretability analysis, forming the foundation of our perturbation-based explanation framework.Theorem 1 shows that, compared with the random perturbation method of Ribeiro et al. (2016) and the clustering-based perturbation method of Zafar and Khan (2021), the proposed fixed-ratio perturbation scheme yields more stable explanations (see Appendix).

Algorithm 1 implements a standardized *perturbation–prediction* construction procedure to quantify how different image regions influence the VLM's decision. Given an input image x, the algorithm first applies the segmentation operator $\mathcal{S}$ (superpixel segmentation) to obtain a structured region set $\mathcal{R} = \{r_1, \ldots, r_L\}$. It then adopts a fixed-percentage masking strategy: for each perturbation $k = 1, \ldots, K$, a binary mask $z^{(k)} \in \{0, 1\}^L$ is constructed under the equality constraint $\sum_{\ell=1}^{L}(1 - z_\ell^{(k)}) = m$, where $m = \lfloor \rho L \rfloor$ and ρ is fixed to 0.3. This design enforces that each perturbed sample masks approximately 30% of the regions, ensuring a consistent perturbation strength across all samples. The masking operator $\mathcal{M}$ is subsequently applied to the segmented image to generate the perturbed image set $\mathcal{X}_{\text{pert}}$.

In the prediction stage, the algorithm queries the VLM f_{VLM} using the same text description t and a unified prompting template for both the original image and each perturbed image. This yields the original prediction $p^{(0)}$ and the perturbed predictions $\{p^{(k)}\}_{k=1}^{K}$, which are aggregated into the prediction vector $\mathbf{p} = (p^{(0)}, p^{(1)}, \ldots, p^{(K)})$. Finally, the algorithm stores each perturbation triplet $(z^{(k)}, x_{\text{pert}}^{(k)}, p^{(k)})$ into a perturbation database $\mathcal{B}$, resulting in a perturbation–prediction dataset that supports subsequent weighting and interpretability modeling. Because the masking ratio is kept constant across perturbations, the constructed dataset more stably captures the mapping from *regional modifications* to *prediction changes*, providing a reliable foundation for region-importance estimation.

Algorithm 1 Perturb samples and prediction generation (Fixed-percentage Masking).

Require: Image x; segmentation operator $\mathcal{S}$; masking operator $\mathcal{M}$; VLM $f_{\text{VLM}}(\cdot, \cdot)$; selected text $t \in \mathcal{D}_{\text{Wiki}}$; mask ratio ρ (fixed, e.g., $\rho = 0.3$); number of perturbations K.

Ensure: Regions $\mathcal{R} = \{r_1, \ldots, r_L\}$; masks $\mathcal{Z} = \{z^{(1)}, \ldots, z^{(K)}\}$; perturbed set $\mathcal{X}_{\text{pert}}$; predictions $\mathbf{p} = (p^{(0)}, p^{(1)}, \ldots, p^{(K)})$; perturbation database $\mathcal{B} = \{(z^{(k)}, x_{\text{pert}}^{(k)}, p^{(k)})\}_{k=1}^{K}$.

Step 1: Segmentation
$x^{\text{seg}} \leftarrow \mathcal{S}(x)$
Decompose x^{seg} into regions $\mathcal{R} = \{r_1, r_2, \ldots, r_L\}$
Step 2: Fixed-percentage Masking (ratio fixed at ρ)
Initialize $\mathcal{Z} \leftarrow \emptyset$, $\mathcal{X}_{\text{pert}} \leftarrow \emptyset$
Let $m \leftarrow \lfloor \rho L \rfloor$ ▷ number of masked regions, fixed
for $k \leftarrow 1$ to K **do**
  Construct $z^{(k)} \in \{0, 1\}^L$ such that $\sum_{\ell=1}^{L}(1 - z_\ell^{(k)}) = m$
  $\mathcal{Z} \leftarrow \mathcal{Z} \cup \{z^{(k)}\}$
  $x_{\text{pert}}^{(k)} \leftarrow \mathcal{M}\left(x^{\text{seg}}, z^{(k)}\right)$
  $\mathcal{X}_{\text{pert}} \leftarrow \mathcal{X}_{\text{pert}} \cup \{x_{\text{pert}}^{(k)}\}$
end for
Step 3: Prediction Generation (Unified template)
Template: "You are now a classifier. Please tell me the probability that the image matches the given text.
The probability is between 0 and 1, rounded to two decimal places."
$p^{(0)} \leftarrow f_{\text{VLM}}(x, t)$
for $k \leftarrow 1$ to K **do**
  $p^{(k)} \leftarrow f_{\text{VLM}}\left(x_{\text{pert}}^{(k)}, t\right)$
end for
$\mathbf{p} \leftarrow (p^{(0)}, p^{(1)}, \ldots, p^{(K)})$
Step 4: Perturbation Database Construction
Initialize $\mathcal{B} \leftarrow \emptyset$
for $k \leftarrow 1$ to K **do**
  $\mathcal{B} \leftarrow \mathcal{B} \cup \{(z^{(k)}, x_{\text{pert}}^{(k)}, p^{(k)})\}$
end for
return $\mathcal{R}$, $\mathcal{Z}$, $\mathcal{X}_{\text{pert}}$, $\mathbf{p}$, $\mathcal{B}$

3.2 The weight for the original image and perturb samples

The purpose of this step is to assign different weights to the perturbed samples according to their similarity to the original image, ensuring that samples more similar to the original image exert greater influence during the subsequent model fitting. As illustrated in Fig. 3, we first compute the cosine

similarity between the original image and each perturbed sample to quantify their distance in the feature space. The similarity values are then transformed into weights through a kernel function, giving higher weights to samples that lie closer to the original image while suppressing those that are farther away. This distance-based weighting strategy effectively controls the contribution of the local neighborhood and enables the model to more stably capture the local decision structure around the original image. The detailed computation is presented as follows:

Since the segmented image x^{seg} can be regarded as a vector composed of L regions, and each perturbed image $x_{\text{pert}}^{(k)}$ is determined by the binary mask $z^{(k)} = (z_1^{(k)}, \ldots, z_L^{(k)})$, a basic similarity measure can be defined as the proportion of retained regions:

$$\text{sim}(x, x_{\text{pert}}^{(k)}) = \frac{1}{L} \sum_{\ell=1}^{L} z_\ell^{(k)}, \tag{13}$$

where $z_\ell^{(k)} \in \{0, 1\}$ indicates whether region r_ℓ is preserved.

However, this similarity changes linearly and does not capture the nonlinear decay commonly observed in perturbation-based analysis. To address this limitation, we introduce an adjusted similarity measure:

$$\text{adj}_k = \exp\big(\lambda\big(1 - \text{sim}(x, x_{\text{pert}}^{(k)})\big)\big), \tag{14}$$

where λ is a hyperparameter controlling the strength of the adjustment, and $1 - \text{sim}(\cdot)$ corresponds to the normalized Hamming distance between the masks.

This nonlinear transformation assigns higher weights to perturbed samples that are closer to the original image and progressively suppresses the influence of samples with larger coverage. Finally, the adjusted similarity is applied to each region of the perturbed image:

$$r'_{\ell,k} = \text{adj}_k \, r_{\ell,k}, \tag{15}$$

where $r_{\ell,k}$ denotes the original value of region r_ℓ in the k-th perturbed image, and $r'_{\ell,k}$ is the updated, weight-adjusted representation. Theorem 2 aims to illustrate the importance of the sensitivity of the adjusted similarity. (see Appendix)

Algorithm 2 describes the weighting strategy that emphasizes perturbed samples that are more similar to the original image, so that the subsequent surrogate model focuses on the local neighborhood of the input. Given the superpixel regions $\mathcal{R} = \{r_1, \ldots, r_L\}$ and the corresponding binary masks $\mathcal{Z}$, the algorithm first computes a base similarity for each perturbation as the retained-region proportion $\text{sim}_k = \frac{1}{L} \sum_{\ell=1}^{L} z_\ell^{(k)}$. To avoid a purely linear decay with respect to the masking level, it then applies an exponential kernel to obtain a nonlinear adjusted similarity weight $\text{adj}_k = \exp\big(\lambda(1 - \text{sim}_k)\big)$,

Algorithm 2 Weighting for the original image and perturbed samples.

Require: Segmented regions $\mathcal{R} = \{r_1, \ldots, r_L\}$; perturbed masks $\mathcal{Z} = \{z^{(1)}, \ldots, z^{(K)}\}$ where $z^{(k)} = (z_1^{(k)}, \ldots, z_L^{(k)}) \in \{0, 1\}^L$; region values $\{r_{\ell,k}\}$ for each perturbed sample k; hyperparameter λ.

Ensure: Adjusted similarity weights $\{\text{adj}_k\}_{k=1}^K$; weighted region representations $\{r'_{\ell,k}\}$; weighted feature vectors $\{\mathbf{X}_k\}_{k=1}^K$.

Step 1: Compute base similarity (retained-region proportion)
for $k \leftarrow 1$ to K **do**
  $\text{sim}_k \leftarrow \frac{1}{L} \sum_{\ell=1}^{L} z_\ell^{(k)}$
end for
Step 2: Nonlinear adjustment (exponential kernel)
for $k \leftarrow 1$ to K **do**
  $\text{adj}_k \leftarrow \exp\big(\lambda\,(1 - \text{sim}_k)\big)$
end for
Step 3: Apply weights to region representations
for $k \leftarrow 1$ to K **do**
  for $\ell \leftarrow 1$ to L **do**
    $r'_{\ell,k} \leftarrow \text{adj}_k \cdot r_{\ell,k}$
  end for
  $\mathbf{X}_k \leftarrow (r'_{1,k}, r'_{2,k}, \ldots, r'_{L,k})$
end for
return $\{\text{adj}_k\}_{k=1}^K$, $\{r'_{\ell,k}\}$, $\{\mathbf{X}_k\}$

where λ controls the strength of the decay. Finally, the weight adj_k is multiplied to every region value $r_{\ell,k}$ in the k-th perturbed sample to form the weighted representation $r'_{\ell,k} = \text{adj}_k \, r_{\ell,k}$, and the resulting weighted feature vector $\mathbf{X}_k = (r'_{1,k}, \ldots, r'_{L,k})$ is returned for downstream parallel-tree training.

3.3 Ensemble parallel-tree

After obtaining the nonlinearly weighted region representations $r'_{\ell,k}$, we proceed to construct the proposed *ensemble parallel tree framework*. As shown in Fig. 4, unlike conventional tree-based interpretable models that rely on Bagging or Boosting (Lee et al. 2025; Jiang et al. 2024), all trees in the ensemble are trained independently and simultaneously on the same perturbation–prediction dataset. This design eliminates repeated resampling and sequential weak learner dependency, thereby reducing computational complexity and improving interpretive stability.

For each perturbed image $x_{\text{pert}}^{(k)}$, the weighted regional vector

$$\mathbf{X}_k = \big(r'_{1,k}, r'_{2,k}, \ldots, r'_{L,k}\big) \tag{16}$$

serves as the input feature representation. Let the ensemble consist of

$$\mathcal{T} = \left\{T^{(1)}, T^{(2)}, \ldots, T^{(S)}\right\}, \tag{17}$$

where S denotes the number of parallel trees.

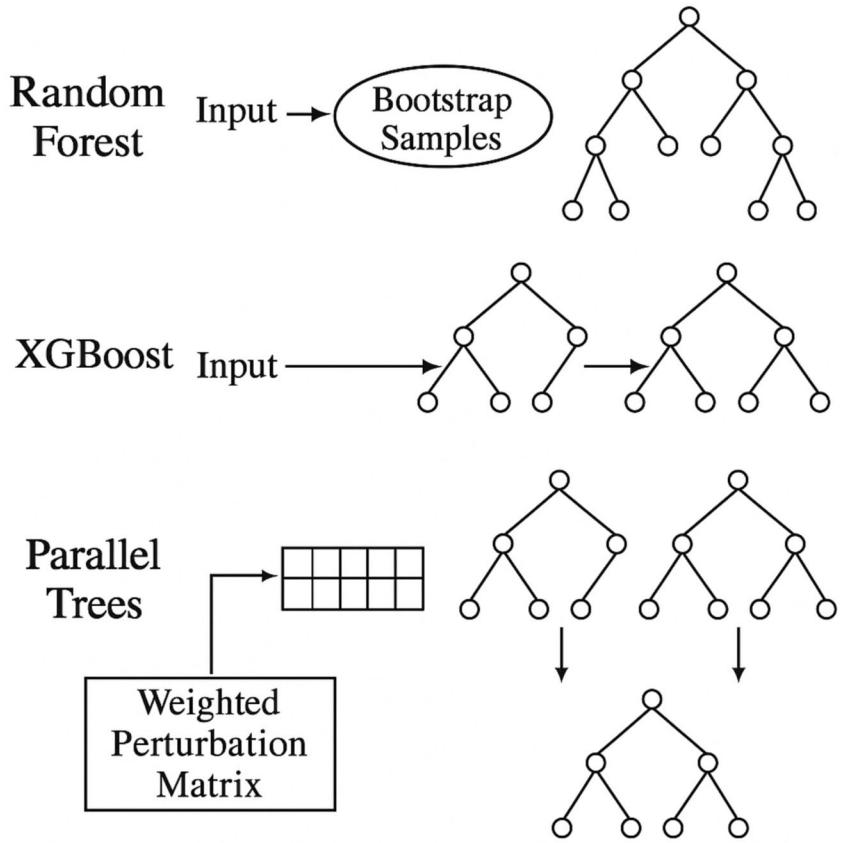


Fig. 4 A comparison of the designed parallel trees with Random Forest and XGBoost

Each tree recursively identifies the optimal split based on the minimization of weighted post–split variance:

$$\left(r_{\ell^*,k}, \tau^*\right) = \arg\min_{r_{\ell,k},\tau} \left[w_{\mathrm{L}} \operatorname{Var}(D_{\mathrm{L}}) + w_{\mathrm{R}} \operatorname{Var}(D_{\mathrm{R}})\right], \quad (18)$$

where D_{L} and D_{R} denote the left and right subsets formed by threshold τ, and w_{L}, w_{R} are their corresponding normalized weights.

Each leaf node outputs the average prediction of the samples assigned to that node:

$$\hat{p}_{\mathrm{leaf}} = \frac{1}{|D_{\mathrm{leaf}}|} \sum_{i \in D_{\mathrm{leaf}}} p(i). \quad (19)$$

The contribution of region r_ℓ is quantified by aggregating the variance reductions across all trees:

$$\mathrm{Imp}(r_\ell) = \frac{1}{S} \sum_{s=1}^{S} \sum_{t \in T^{(s)}} \Delta \mathrm{Var}_t(r_\ell), \quad (20)$$

where $\Delta \mathrm{Var}_t(r_\ell)$ denotes the reduction in variance at node t when region r_ℓ is selected as the splitting feature.

To ensure robustness under different perturbations, a structural regularization term is applied to each tree:

$$\Omega(T^{(s)}) = \gamma L^{(s)} + \frac{\lambda}{2} \left\| w^{(s)} \right\|^2, \quad (21)$$

where $L^{(s)}$ is the number of leaves in tree $T^{(s)}$, γ penalizes excessive tree growth, and λ regulates the magnitude of leaf predictions. This regularization effectively suppresses model sensitivity and ensures stable and consistent explanations across perturbations. Theorem 3 aims to demonstrate that the designed parallel tree method has a lower complexity compared to models that use Random Forest or XGBoost as the interpretability module. (see Appendix)

Algorithm 3 presents the proposed *ensemble parallel-tree* framework built on the perturbation–prediction dataset. Given the weighted region-level feature vectors $\mathbf{X}_k = (r'_{1,k}, \ldots, r'_{L,k})$ and their corresponding VLM predictions $p^{(k)}$, the method initializes an ensemble $\mathcal{T} = \{T^{(1)}, \ldots, T^{(S)}\}$ and trains all S trees independently on the same dataset. For each tree, the splitting process is performed recursively until a maximum depth (or another stopping rule) is reached. At each node, the optimal split feature and threshold (ℓ^*, τ^*) are selected by minimizing the weighted post-split variance $w_{\mathrm{L}}\mathrm{Var}(D_{\mathrm{L}}) + w_{\mathrm{R}}\mathrm{Var}(D_{\mathrm{R}})$, where D_{L} and D_{R} denote the

Algorithm 3 Ensemble parallel-tree.

Require: Weighted feature vectors $\{\mathbf{X}_k\}_{k=0}^{K}$ with $\mathbf{X}_k = (r'_{1,k}, \ldots, r'_{L,k})$; predictions $\{p^{(k)}\}_{k=0}^{K}$; number of trees S; max depth $d_{\max}$ (or other stopping rule); regularization parameters (γ, λ).

Ensure: Trained ensemble $\mathcal{T} = \{T^{(1)}, \ldots, T^{(S)}\}$; region importance scores $\{\mathrm{Imp}(r_\ell)\}_{\ell=1}^{L}$.

Step 1: Build the parallel-tree ensemble
Initialize $\mathcal{T} \leftarrow \{T^{(1)}, T^{(2)}, \ldots, T^{(S)}\}$
for $s \leftarrow 1$ to S **do**
  Train tree $T^{(s)}$ independently on $\mathcal{D} = \{(\mathbf{X}_k, p^{(k)})\}_{k=0}^{K}$
  Grow $T^{(s)}$ recursively until reaching $d_{\max}$ or a stopping criterion
  while a split is allowed at the current node **do**
    Search over candidate features $r'_{\ell,k}$ and thresholds τ
    Choose (ℓ^*, τ^*) minimizing the weighted post-split variance: $(\ell^*, \tau^*) \leftarrow \arg\min_{\ell,\tau} \left[w_{\mathrm{L}} \mathrm{Var}(D_{\mathrm{L}}) + w_{\mathrm{R}} \mathrm{Var}(D_{\mathrm{R}}) \right]$
    Split data into $D_{\mathrm{L}} = \{k : r'_{\ell^*,k} \le \tau^*\}$ and $D_{\mathrm{R}} = \{k : r'_{\ell^*,k} > \tau^*\}$
  end while
  For each leaf D_{leaf}, set the leaf output: $\hat{p}_{\mathrm{leaf}} \leftarrow \frac{1}{|D_{\mathrm{leaf}}|} \sum_{i \in D_{\mathrm{leaf}}} p(i)$
  Apply structural regularization: $\Omega(T^{(s)}) \leftarrow \gamma L^{(s)} + \frac{\lambda}{2} \|w^{(s)}\|^2$
end for
Step 2: Compute region importance by variance reduction aggregation
for $\ell \leftarrow 1$ to L **do**
  $\mathrm{Imp}(r_\ell) \leftarrow 0$
end for
for $s \leftarrow 1$ to S **do**
  for all split nodes $t \in T^{(s)}$ **do**
    Let feature used at node t be region index $\ell(t)$
    Compute variance reduction $\Delta\mathrm{Var}_t(r_{\ell(t)})$
    $\mathrm{Imp}(r_{\ell(t)}) \leftarrow \mathrm{Imp}(r_{\ell(t)}) + \Delta\mathrm{Var}_t(r_{\ell(t)})$
  end for
end for
for $\ell \leftarrow 1$ to L **do**
  $\mathrm{Imp}(r_\ell) \leftarrow \frac{1}{S} \mathrm{Imp}(r_\ell)$
end for
return $\mathcal{T}, \{\mathrm{Imp}(r_\ell)\}_{\ell=1}^{L}$

left/right subsets and w_{L}, w_{R} are normalized subset weights. Leaf nodes output the mean prediction of samples assigned to the leaf, and a structural regularization term $\Omega(T^{(s)}) = \gamma L^{(s)} + \frac{\lambda}{2} \|w^{(s)}\|^2$ is applied to control tree complexity and reduce sensitivity.

After training, the algorithm computes region importance scores by aggregating variance reductions across the entire ensemble. Specifically, whenever a region r_ℓ is used as the splitting feature at a node t, the corresponding variance reduction $\Delta\mathrm{Var}_t(r_\ell)$ is accumulated into its importance. Summing these contributions over all split nodes and all trees yields an overall importance measure, which is then averaged by the number of trees: $\mathrm{Imp}(r_\ell) = \frac{1}{S} \sum_{s=1}^{S} \sum_{t \in T^{(s)}} \Delta\mathrm{Var}_t(r_\ell)$. This design differs from bagging- or boosting-based ensembles (e.g., Random Forest or XGBoost) because it removes resampling and sequential dependency, enabling parallel training while producing stable, consistent region-importance estimates for downstream visualization and global explanation.

3.4 Important block output

After the ensemble parallel trees have been fully trained, we evaluate the importance of each region and visualize the areas that the model regards as most influential. Specifically, for each tree T_s $(1 \le s \le S)$, we compute the contribution of region r_ℓ based on the reduction in variance induced by splits involving that region. Let $\mathrm{Imp}_s(r_\ell)$ denote the importance of region r_ℓ measured from tree T_s. The aggregated importance across the ensemble is then defined as:

$$\mathrm{Imp}(r_\ell) = \frac{1}{S} \sum_{s=1}^{S} \mathrm{Imp}_s(r_\ell), \qquad 1 \le \ell \le L. \tag{22}$$

We rank all regions according to their aggregated importance $\mathrm{Imp}(r_\ell)$ and select the top-ranked regions for visualization. These important regions are subsequently mapped back to the spatial layout of the original image and combined with the weight-adjusted representation $r'_{\ell,k}$ to generate a final saliency map that highlights the locations most relied upon by the model during inference. This process not only provides a spatially intuitive explanation but also further validates the rationality and stability of the proposed perturbation and weighting strategies.

Algorithm 4 describes how the proposed method produces *important block output* by converting the trained ensemble parallel trees into a region-level ranking and an image-space visualization. Given the ensemble $\mathcal{T} = \{T^{(1)}, \ldots, T^{(S)}\}$ and the segmented regions $\mathcal{R} = \{r_1, \ldots, r_L\}$, the algorithm first initializes the aggregated importance scores $\mathrm{Imp}(r_\ell)$ for all regions. It then traverses every split node in every tree and accumulates the variance reduction $\Delta\mathrm{Var}_t(r_\ell)$ whenever region r_ℓ is selected as the splitting feature. After summing contributions across all trees, the importance scores are normalized by the number of trees, yielding the final aggregated importance $\mathrm{Imp}(r_\ell)$ that reflects how strongly each region contributes to reducing prediction uncertainty in the ensemble.

Based on these aggregated scores, the algorithm ranks all regions in descending order and selects the top-m regions as the most influential blocks, denoted by $\mathcal{M}_{\mathrm{top}}$. To generate an intuitive explanation, each selected region is mapped back to its spatial location in the original image, and a saliency map $\mathcal{H}$ is constructed on the image grid by assigning region-level importance values (or optionally the weight-adjusted representation $r'_{\ell,k}$) to the corresponding pixels. The resulting saliency map highlights the locations most relied upon by the model during inference, providing an interpretable

Algorithm 4 Important block output (Region Ranking and Saliency Map).

Require: Trained ensemble $\mathcal{T} = \{T^{(1)}, \ldots, T^{(S)}\}$; regions $\mathcal{R} = \{r_1, \ldots, r_L\}$ (with spatial layout); variance-reduction records per tree node (or computable $\Delta\mathrm{Var}_t$); top-m; (optional) weighted region representations $r'_{\ell,k}$ for visualization.

Ensure: Aggregated importance $\{\mathrm{Imp}(r_\ell)\}_{\ell=1}^{L}$; selected important regions $\mathcal{M}_{\mathrm{top}}$; saliency map $\mathcal{H}$ on the original image.

Step 1: Compute tree-level and aggregated region importance
for $\ell \leftarrow 1$ to L **do**
  $\mathrm{Imp}(r_\ell) \leftarrow 0$
end for
for $s \leftarrow 1$ to S **do**
  for all split nodes $t \in T^{(s)}$ **do**
    Let the splitting feature at node t correspond to region index $\ell(t)$
    Compute variance reduction $\Delta\mathrm{Var}_t(r_{\ell(t)})$
    $\mathrm{Imp}(r_{\ell(t)}) \leftarrow \mathrm{Imp}(r_{\ell(t)}) + \Delta\mathrm{Var}_t(r_{\ell(t)})$
  end for
end for
for $\ell \leftarrow 1$ to L **do**
  $\mathrm{Imp}(r_\ell) \leftarrow \frac{1}{S}\,\mathrm{Imp}(r_\ell)$
end for
Step 2: Select the top-m important regions
Rank regions $\{r_\ell\}_{\ell=1}^{L}$ by descending $\mathrm{Imp}(r_\ell)$
$\mathcal{M}_{\mathrm{top}} \leftarrow$ top-m regions by $\mathrm{Imp}(r_\ell)$
Step 3: Generate saliency map in image space
Initialize saliency map $\mathcal{H} \leftarrow \mathbf{0}$ on the image grid
for all $r_\ell \in \mathcal{M}_{\mathrm{top}}$ **do**
  Project region r_ℓ back to its pixel coordinates in the original image
  Update $\mathcal{H}$ on pixels of r_ℓ using $\mathrm{Imp}(r_\ell)$ (or $r'_{\ell,k}$ if used)
end for
return $\{\mathrm{Imp}(r_\ell)\}_{\ell=1}^{L}$, $\mathcal{M}_{\mathrm{top}}$, $\mathcal{H}$

visualization that is consistent with the variance-based importance measure.

3.5 Global explain

After obtaining the region-level importance scores and visualizations for each individual image, we further aggregate these results over the entire dataset to derive a robust global metric. Although prior studies (Jiang et al. 2025; Lee et al. 2025) have proposed dataset-level evaluation procedures, their approaches typically rely on simple weighted averages, which are highly sensitive to outliers and may lead to distorted global indicators.

To overcome this limitation, we adopt the trimmed mean as the global aggregation strategy. Compared with conventional averaging, the Trimmed Mean effectively suppresses the influence of extreme cases and improves robustness, which is particularly important in medical imaging where large inter-case variability is common.

Let each image provide the top-m important regions selected by both the expert and the model. Define:

- E_i: expert-annotated important regions of image i,
- M_i: model-predicted important regions of image i,
- $|E_i \cap M_i|$: the number of overlapping regions.

The matching rate for image i is computed as

$$\alpha_i = \frac{|E_i \cap M_i|}{m}. \tag{23}$$

Sorting all N matching rates in ascending order gives

$$\alpha_{(1)} \le \alpha_{(2)} \le \cdots \le \alpha_{(N)}. \tag{24}$$

Given a trimming parameter k (e.g., $k = 0.1N$), we discard the smallest k and largest k values and compute the trimmed mean:

$$\mathrm{TrimmedMean}_k(\alpha) = \frac{1}{N-2k} \sum_{t=k+1}^{N-k} \alpha_{(t)}. \tag{25}$$

The resulting value is reported as the global explanation consistency index, providing a stable and noise-resistant measure of how consistently the model agrees with expert annotations across the dataset. In Theorem 3, we explained that the trimmed mean method is fairer than the aggregate mean method.

Algorithm 5 Global explain via trimmed mean consistency index.

Require: Dataset size N; top-m; trimming parameter k; expert-selected important regions $\{E_i\}_{i=1}^{N}$; model-selected important regions $\{M_i\}_{i=1}^{N}$ (top-m per image, e.g., from Algorithm 4).

Ensure: Global explanation consistency index $\mathrm{TrimmedMean}_k(\alpha)$.

Step 1: Compute matching rates for each image
for $i \leftarrow 1$ to N **do**
  $\alpha_i \leftarrow \frac{|E_i \cap M_i|}{m}$
end for
Step 2: Sort matching rates
Sort $\{\alpha_i\}_{i=1}^{N}$ in ascending order to obtain $\alpha_{(1)} \le \alpha_{(2)} \le \cdots \le \alpha_{(N)}$
Step 3: Trim extremes and compute trimmed mean
Discard the smallest k and largest k values
$\mathrm{TrimmedMean}_k(\alpha) \leftarrow \frac{1}{N-2k} \sum_{t=k+1}^{N-k} \alpha_{(t)}$
return $\mathrm{TrimmedMean}_k(\alpha)$

Algorithm 5 summarizes the proposed *global explain* procedure that aggregates image-level explanation results into a robust dataset-level consistency metric. For each image $i \in \{1, \ldots, N\}$, the algorithm compares the top-m regions selected by the expert, E_i, with the top-m regions predicted by the model, M_i, and computes a matching rate $\alpha_i = \frac{|E_i \cap M_i|}{m}$. This per-image score measures how well the model-identified important blocks align with expert annotations, providing a normalized agreement value in [0, 1] for every case.

To reduce sensitivity to outliers and extreme cases, the algorithm then sorts the matching rates into $\alpha_{(1)} \leq \alpha_{(2)} \leq \cdots \leq \alpha_{(N)}$ and applies a trimmed-mean aggregation. Specifically, it discards the smallest k and largest k values and computes $\text{TrimmedMean}_k(\alpha) = \frac{1}{N-2k} \sum_{t=k+1}^{N-k} \alpha_{(t)}$. The resulting statistic is reported as the global explanation consistency index, yielding a stable and noise-resistant measure of how consistently the model agrees with expert-selected regions across the entire dataset.

The above concludes all parts of the proposed MedExplainer; next, we will introduce the experimental section.

4 Experiments and performance evaluations

This section presents the experimental setup and performance evaluation of this paper. It is organized into the following parts: dataset and experimental settings, evaluation metrics, hyperparameter settings, comparison with state-of-the-art (SOTA) models, ablation study, and case Study.

4.1 Dataset and experimental settings

This section describes the datasets and computational resources used in our experiments. To evaluate the effectiveness of the proposed method, we consider two types of data: conventional static medical images and surgical video recordings. In the benchmark evaluation, we conduct assessment and validation using standard medical imaging data. In the case study section, we further investigate the application of the proposed interpretable machine learning method on selected medical surgical videos.

For the static-image experiments, we employ the Radiology Objects in COntext (ROCO) dataset (Pelka et al. 2018), the UDIAT breast ultrasound dataset (Yap et al. 2017), and the COVID-19 CT Scans dataset (Rahimzadeh et al. 2021).

ROCO is a large-scale multimodal medical image corpus containing more than 81,000 radiology images paired with corresponding clinical text descriptions, covering imaging modalities such as X-ray, CT, and MRI (Pelka et al. 2018). It is widely used to evaluate multimodal understanding and vision–language alignment in medical AI systems.

The UDIAT dataset consists of clinically acquired breast ultrasound images with expert-annotated lesion masks and is commonly used for lesion detection and benchmarks in medical image analysis (Yap et al. 2017).

The COVID-19 CT Scans dataset is a large chest CT imaging dataset for COVID-19 research, containing tens of thousands of CT images from hundreds of patients. For example, the widely used COVID-CTset variant includes on the order of 60,000+ axial CT slices from 377 subjects, with both COVID-19 positive and normal scans available for training and evaluation, and expert-annotated lung and infection segmentation masks where applicable (Rahimzadeh et al. 2021).

The complete dataset is partitioned into three mutually exclusive subsets: a training set, a validation set, and a test set. The splitting ratio is set to $7:2:1$, corresponding to 70% of the data for training, 20% for validation, and 10% for testing. The test set is held out and used exclusively for final performance evaluation, while the training and validation sets are utilized during model development and hyperparameter tuning.

To ensure robustness and reduce sampling bias, an X-fold cross-validation strategy is employed on the training and validation data. In our experiments, the number of folds is set to $X = 5$. Specifically, the training and validation data are divided into five equally sized folds; in each iteration, four folds are used for training and one fold is used for validation. The final validation performance is reported as the average over all folds.

Only samples that meet the predefined quality and completeness requirements are included in the dataset. Samples with missing critical information, corrupted files, or annotation errors are excluded from the experiments. In addition, duplicate or near-duplicate samples are removed to prevent data leakage across different subsets.

All inclusion and exclusion criteria are applied prior to dataset splitting to ensure that no invalid samples appear in any of the training, validation, or test sets.

Before model training, all data undergo a unified preprocessing pipeline. This includes data normalization, format standardization, and, where applicable, noise removal. Preprocessing operations are applied consistently across all subsets to maintain distributional consistency.

Importantly, preprocessing parameters are computed exclusively on the training data and then applied to the validation and test sets to avoid information leakage. This ensures a fair and unbiased evaluation of the proposed method.

For evaluation, each image is partitioned into blocks using the *scikit-image* toolkit according to the predefined grid structure. The predictions produced by our model are then compared against the expert-provided segmentation masks through block-level overlap analysis. For video-based experiments, we use bladder cancer surgical recordings collected at a major medical center in Asia. All video data were fully anonymized.

The large vision models (VLMs) used for interpretability include Google's open-source Medical-Gemma-4B model and the Gemini model. To ensure fairness, the open-source medical imaging model is further adapted using LoRA fine-tuning on the relevant datasets, allowing it to better handle the imaging domains involved in our evaluation. All experiments are conducted on NVIDIA A100 80GB GPU.

4.2 Evaluation metrics

We evaluate the feasibility of different interpretability methods from three perspectives: **stability**, **fidelity**, and **global interpretability**. In addition, to ensure a meaningful correspondence between the model's interpretable localization results and the actual diagnostic categories (i.e., lesion tissue versus normal tissue), we perform evaluation based on regions of interest (ROIs) and introduce three metrics, including ROI Coverage, Intersection over Union (IoU), Pointing Game, and the Dice Similarity Coefficient (Dice) for quantitative analysis.

The corresponding mathematical definitions are consistent with the notation introduced in previous sections.

To assess stability, we adopt the Jaccard Index to quantify the consistency of important-region selections across repeated trials or under different perturbations. Let $\mathcal{I}^{(a)}$ and $\mathcal{I}^{(b)}$ denote the sets of important regions identified in experiment a and b, respectively. The Jaccard Index is defined as

$$\text{Jaccard}\left(\mathcal{I}^{(a)}, \mathcal{I}^{(b)}\right) = \frac{\left|\mathcal{I}^{(a)} \cap \mathcal{I}^{(b)}\right|}{\left|\mathcal{I}^{(a)} \cup \mathcal{I}^{(b)}\right|}. \tag{26}$$

The overall stability score is obtained by averaging over all pairs of experiments:

$$\text{Stability} = \frac{1}{M} \sum_{(a,b)} \text{Jaccard}\left(\mathcal{I}^{(a)}, \mathcal{I}^{(b)}\right), \tag{27}$$

where M denotes the total number of experiment pairs.

Fidelity evaluates how accurately the explanation model (e.g., the parallel tree ensemble) reconstructs the prediction behavior of the VLM. Let $p(k)$ be the prediction of the VLM for the k-th perturbed sample and $\hat{p}^{(k)}$ be the output of the explanation model. The coefficient of determination is given by

$$R^2 = 1 - \frac{\sum_{k=1}^{K} \left(p(k) - \hat{p}^{(k)}\right)^2}{\sum_{k=1}^{K} \left(p(k) - \bar{p}\right)^2}, \tag{28}$$

where $\bar{p} = \frac{1}{K} \sum_{k=1}^{K} p(k)$.

Considering the model complexity caused by S trees, we compute the Adjusted R^2:

$$\text{Adj}R^2 = 1 - \frac{(1 - R^2)(K - 1)}{K - S - 1}. \tag{29}$$

Higher values of $\text{Adj}R^2$ indicate that the explanation model faithfully reproduces the VLM's decision structure.

Global interpretability is evaluated using the unified aggregation method described in Section 3.5, based on the Trimmed Mean. Given a set of region-importance scores $\{\alpha_1, \ldots, \alpha_n\}$, we first sort them as $\alpha_{(1)} \le \alpha_{(2)} \le \cdots \le \alpha_{(n)}$. After removing the lowest and highest k values, the Trimmed Mean is defined as:

$$\text{TrimmedMean}_k = \frac{1}{n - 2k} \sum_{t=k+1}^{n-k} \alpha_{(t)}. \tag{30}$$

This robust aggregation suppresses the effect of outliers or extreme samples, resulting in a more stable global interpretability measure.

For region-of-interest (ROI) evaluation, we introduce three metrics, namely ROI Coverage, Intersection over Union (IoU), and the Pointing Game.

ROI Coverage measures how well the predicted explanation region covers the ground-truth ROI:

$$\text{Coverage} = \frac{|R_{\text{pred}} \cap R_{\text{gt}}|}{|R_{\text{gt}}|}. \tag{31}$$

Intersection over Union (IoU) evaluates the spatial overlap between the predicted region and the ground-truth ROI:

$$\text{IoU} = \frac{|R_{\text{pred}} \cap R_{\text{gt}}|}{|R_{\text{pred}} \cup R_{\text{gt}}|}. \tag{32}$$

Pointing Game is a point-based localization metric. Let

$$p^* = \arg\max_p H(p) \tag{33}$$

denote the pixel with the maximum response in the explanation heatmap. A hit is defined as

$$\text{Hit}(i) = \begin{cases} 1, & p_i^* \in R_{\text{gt}} \\ 0, & \text{otherwise} \end{cases} \tag{34}$$

and the final Pointing Game score is computed as

$$\text{PointingGame} = \frac{1}{N} \sum_{i=1}^{N} \text{Hit}(i). \tag{35}$$

Dice Similarity Coefficient (Dice) is a widely used overlap-based metric for evaluating segmentation quality and region-level agreement between predicted regions and ground-truth annotations. It measures the degree of spatial overlap by jointly considering the intersection and the sizes of both regions.

Given a predicted region R_{pred} and the corresponding ground-truth region R_{gt}, the Dice coefficient is defined as:

$$\text{Dice} = \frac{2\,|R_{\text{pred}} \cap R_{\text{gt}}|}{|R_{\text{pred}}| + |R_{\text{gt}}|}. \tag{36}$$

The Dice score ranges from 0 to 1, where a value of 1 indicates perfect spatial agreement between the predicted and

ground-truth regions, and a value of 0 indicates no overlap. Compared with Intersection over Union (IoU), Dice places greater emphasis on the overlapping region and is particularly suitable for medical imaging segmentation tasks, especially when the target regions are small or class-imbalanced.

4.3 Hyperparameter settings

In the settings related to perturbation generation, model training, and weight adjustment, the proposed method adopts a unified and reproducible configuration of hyperparameters. During perturbation construction, the input image is first partitioned into a set of structured regions through the superpixel segmentation process defined in (1)–(2). The number of regions is adaptively determined by the SLIC algorithm implemented in the scikit-image toolkit and is maintained in the range of approximately 100 to 150 in practice, balancing spatial expressiveness and computational efficiency. The number of perturbations used to construct the perturbation–prediction database is controlled by the parameter K in (8), which is fixed to 1000 in all experiments to achieve a reasonable trade-off between interpretability stability and computational cost. To avoid explanation instability caused by random masking, the masking ratio parameter in (6) is fixed to 0.3, meaning that approximately 30% of the regions are masked in each perturbed sample. This design ensures consistent perturbation strength across all samples. In implementation, the masking operation corresponding to (7) is realized by setting the pixel values of the selected regions to zero or replacing them with mean values, while keeping the remaining regions unchanged, such that perturbations affect only the spatial structure without introducing additional noise.

During the construction of the explanation model, the number of parallel decision trees defined in (17) is uniformly set to 50. This setting effectively reduces randomness caused by structural variations of individual trees while avoiding unnecessary computational overhead. The maximum depth of each tree is limited to 5, or tree growth is terminated early when the number of samples falls below a predefined threshold, in order to prevent overfitting to perturbed samples. In addition, the structural regularization term in (21) is employed to constrain tree complexity, where the parameter penalizing the number of leaf nodes is set to 0.1, thereby suppressing excessive tree growth and improving the stability of the explanations.

In the perturbation weighting stage, the kernel hyperparameter in (14) is used to control the strength of similarity decay. Ablation studies indicate that when this parameter is set too small, the weighting effect becomes insufficient and explanations are susceptible to noise, whereas overly large values excessively emphasize local perturbations and weaken global consistency. Based on experimental results across different datasets and models, the parameter in (14) is finally fixed to 0.3 and kept constant throughout all experiments.

In the inference setup of the Vision–Language Model (VLM), we adopt a unified querying strategy for all original and perturbed samples, following the fixed prompting scheme described in Algorithm 1. This design ensures that every sample is evaluated under identical inference conditions, such that the resulting prediction scores and explanation outcomes are directly comparable across different experimental settings.

For each VLM, the query is formulated as a paired input consisting of an image $\mathbf{I}$ and a text prompt $\mathbf{p}(\mathbf{t})$, where a disease description $\mathbf{t}$ is inserted into a fixed instruction template. Concretely, all VLM queries use the same unchanged prompt template: *"You are now a classifier. Please tell me the probability that the image matches the given text. The probability is between 0 and 1, rounded to two decimal places."* This template is kept identical across all datasets, model backbones, and experimental configurations; only the image input $\mathbf{I}$ and the paired textual description $\mathbf{t}$ vary. At inference time, the VLM is queried with the pair $(\mathbf{I}, \mathbf{p}(\mathbf{t}))$, and a single scalar probability $\hat{s} \in [0, 1]$ is decoded from the model output. For output parsing, we extract the first valid numeric value within [0, 1] (rounded to two decimals as instructed). If multiple numbers appear in the response, the first one is selected. In the rare case that the response is not directly parseable, the model is re-queried once using the same template with an additional instruction, *"Output only the number."* This fallback mechanism is applied uniformly across all models to preserve protocol consistency. Together, these steps define the unified evaluation protocol used throughout this work.

For each original image $\mathbf{I}$ and its perturbed counterpart $\tilde{\mathbf{I}}$, the same textual description $\mathbf{t}$ and the same prompt template are used to obtain prediction scores $\hat{s}(\mathbf{I}, \mathbf{t})$ and $\hat{s}(\tilde{\mathbf{I}}, \mathbf{t})$. By holding the prompt and text constant and varying only the image content, the proposed setup eliminates confounding effects introduced by prompt engineering and ensures that any observed change in the prediction score is attributable solely to the image perturbation.

The textual descriptions used in (9) are drawn from a medical disease description database constructed from Wikipedia via an automated web-crawling and cleaning pipeline. First, a list of disease-related keywords is manually curated based on the label space of the experimental datasets. A Python-based crawler then retrieves the corresponding Wikipedia pages through publicly accessible interfaces. During the crawling process, only clinically relevant sections—such as disease definitions, signs and symptoms, diagnostic information, and imaging-related descriptions—are retained, while irrelevant content (e.g., historical background, societal context, or reference lists) is removed. The collected text is subsequently

normalized through redundancy filtering, sentence deduplication, and length trimming to ensure consistent semantic density across disease entries.

To associate each input image with an appropriate textual description, the dataset-provided class label is used as the anchor for retrieval and matching. Specifically, for an image $\mathbf{I}_i$ with ground-truth category y_i, the Wikipedia entry corresponding to y_i (or its canonical synonym) is retrieved from the database and used as the description $\mathbf{t}_i$. When multiple aliases exist for a given label—such as abbreviations, alternative disease names, or spelling variants—the label is mapped to a canonical disease concept using a manually verified alias table constructed during keyword curation. The crawler then retrieves the page associated with this canonical concept. In cases where multiple candidate pages are returned due to ambiguity, the correct page is selected based on title and summary matching, as well as the presence of medically indicative keywords related to radiology or imaging.

For fine-grained datasets in which multiple subtypes share a common parent disease category, both parent-level descriptions and subtype-specific sentences are stored when available. The final description $\mathbf{t}_i$ is selected by prioritizing subtype-specific content; if such content is unavailable, the parent-level description is used as a fallback. This label-anchored retrieval strategy ensures that each image is paired with a clinically relevant and category-consistent description while maintaining a deterministic and reproducible text selection process.

For each perturbed sample $\tilde{\mathbf{I}}_i$ generated from an original image $\mathbf{I}_i$, the same textual description $\mathbf{t}_i$ is reused without modification, guaranteeing that the only difference between the two VLM queries lies in the image input. Each processed description is paired with its corresponding medical image to form the final VLM input, and this identical procedure is applied consistently to all original and perturbed samples.

Through the use of an exact and fixed prompting template, an explicit output-parsing protocol, and a label-anchored, deterministic Wikipedia-based text retrieval and matching process with alias resolution and ambiguity handling, the proposed inference setup ensures explanation stability, reproducibility, and cross-model applicability under a unified evaluation framework.

4.4 Comparison with SOTA models

From the results reported in Table 3, it can be observed that different interpretable methods exhibit substantial performance variations in terms of stability, consistency, and robustness across different Vision–Language Models and medical datasets. Overall, all methods achieve consistently better performance on Medical-Gemma-4B than on Gemini Nano Banana, a trend that holds across both the ROCO and UDIAT datasets as well as the Stability, AdjR^2, and TrimmedMean metrics. This observation indicates that stronger medical-domain pretraining of the backbone model plays a crucial role in improving the effectiveness of downstream interpretability methods. In addition, compared with the ROCO dataset, higher scores are generally obtained on the UDIAT dataset for most methods and metrics, particularly in terms of Stability and TrimmedMean, suggesting that the more clearly defined lesion structures in ultrasound images facilitate the generation of stable and consistent explanations.

From a methodological perspective, early gradient-based or simple perturbation-based approaches (e.g., Zeiler & Fergus and LIME) show relatively limited performance across all three metrics, with particularly large fluctuations in Stability, reflecting their sensitivity to perturbation sampling strategies and model randomness. KernelSHAP and D-LIME improve fitting consistency to some extent, yet they still struggle to ensure stable explanations in complex medical imaging scenarios. In contrast, more recent enhanced or ensemble-based perturbation methods, such as G-LIME, EnEXP, and RealEXP, achieve notable improvements across all metrics, demonstrating that structured perturbation strategies or ensemble mechanisms can effectively mitigate instability caused by single perturbations. Among them, RealEXP approaches the upper performance bound of existing methods in several settings, reaching a Stability score of 0.94 under the Medical-Gemma-4B configuration.

The recently proposed LFP method also demonstrates strong competitiveness across different model and dataset combinations. Under the Medical-Gemma-4B setting, LFP achieves Stability scores of 0.92 and 0.90 on the ROCO and UDIAT datasets, respectively, while maintaining relatively high AdjR^2 and TrimmedMean values, outperforming most baseline methods. Under the Gemini Nano Banana setting, although the overall performance is constrained by the model capacity, LFP still yields relatively stable results on the UDIAT dataset, indicating a certain degree of generalization across models of different scales and inference capabilities. Nevertheless, compared with the best-performing method, LFP still falls short in terms of explanation stability and overall consistency.

Overall, the proposed MedExplainer achieves the best performance across all experimental settings reported in Table 3. On Medical-Gemma-4B, it attains Stability scores of 0.96 and 0.95 on the ROCO and UDIAT datasets, respectively, and consistently outperforms all competing methods in terms of AdjR^2 and TrimmedMean. These results indicate that the surrogate model constructed by MedExplainer more accurately captures the behavior of the original black-box model while remaining robust to perturbation-induced noise. Under the Gemini Nano Banana setting, the proposed method likewise maintains leading performance across all three metrics, further confirming its stability and applicability under

Table 3 Comparison of various interpretable methods when using Medical-Gemma-4B and Gemini Nano Banana on the ROCO and UDIAT datasets

Method	ROCO			UDIAT		
	Stability	AdjR^2	TrimmedMean	Stability	AdjR^2	TrimmedMean
Medical-Gemma-4B						
Zeiler and Fergus (2014)	0.53	0.68	0.58	0.55	0.72	0.66
LIME (Ribeiro et al. 2016)	0.68	0.66	0.64	0.61	0.72	0.64
KernelSHAP (Lundberg and Lee 2017)	0.74	0.81	0.72	0.64	0.73	0.68
D-LIME (Zafar and Khan 2021)	0.85	0.84	0.83	0.76	0.74	0.72
G-LIME (Tan et al. 2024)	0.88	0.86	0.84	0.84	0.80	0.87
EnEXP (Lee et al. 2025)	0.91	0.86	0.81	0.91	0.89	0.84
RealEXP (Jiang et al. 2024)	0.94	0.87	0.88	0.94	0.88	0.93
LFP (Chih et al., 2025)	0.92	0.85	0.88	0.90	0.89	0.92
Proposed MedExplainer	0.96	0.93	0.94	0.95	0.96	0.97
Gemini Nano Banana						
Zeiler and Fergus (2014)	0.51	0.62	0.46	0.53	0.61	0.43
LIME (Ribeiro et al. 2016)	0.62	0.57	0.53	0.53	0.55	0.58
KernelSHAP (Lundberg and Lee 2017)	0.65	0.74	0.61	0.64	0.68	0.64
D-LIME (Zafar and Khan 2021)	0.71	0.78	0.72	0.62	0.75	0.77
G-LIME (Tan et al. 2024)	0.79	0.81	0.64	0.85	0.88	0.78
EnEXP (Lee et al. 2025)	0.81	0.85	0.84	0.87	0.84	0.76
RealEXP (Jiang et al. 2024)	0.85	0.83	0.85	0.86	0.81	0.77
LFP (Chih et al., 2025)	0.77	0.79	0.82	0.85	0.86	0.79
Proposed MedExplainer	0.86	0.88	0.91	0.91	0.93	0.85

different model configurations. Taken together, the experimental results in Table 3 clearly demonstrate that the proposed approach offers significant advantages in terms of stability, consistency, and robustness for interpretable medical image analysis.

As shown in Fig. 5, the proposed MedExplainer demonstrates significantly superior stability across all interpretability metrics compared with existing methods. For each model–dataset combination, the boxplots indicate that MedExplainer consistently exhibits the smallest variance, suggesting that its explanations remain highly robust under repeated perturbations. In contrast, traditional perturbation-based interpretability methods, such as Zeiler & Fergus, LIME, and KernelSHAP, present noticeably wider interquartile ranges and longer whiskers, indicating higher sensitivity to randomness and reduced reliability. This phenomenon is particularly evident on the UDIAT dataset, where the variance differences between MedExplainer and baseline methods become more pronounced.

Furthermore, the maximum values of the distributions are consistent with the performance scores reported in Table 3, further confirming the statistical validity of the experimental results. Notably, MedExplainer not only achieves the highest Stability, AdjR^2, and Trimmed Mean scores, but also exhibits the smallest standard deviation across these metrics, indicating that the observed improvements arise from both enhanced explanatory fidelity and substantially improved consistency. Collectively, these results demonstrate that the proposed method delivers more reliable and reproducible interpretability outcomes, which is particularly critical for medical imaging applications where explanation reliability is essential.

Figure 6 compares the performance stability of Grad-CAM, Increment-CAM, R-CAM, and the proposed MedExplainer on both the ROCO and UDIAT medical imaging datasets. Performance distributions are illustrated using Adj-R^2 and Trimmed Mean metrics. The results demonstrate that MedExplainer achieves substantially higher stability and lower variance across datasets compared to existing gradient-based interpretability methods.

From the results reported in Table 4, it can be observed that different interpretability methods exhibit substantial performance variations on the COVID-19 CT Scans dataset in terms of ROI Coverage, Intersection over Union (IoU), and the Pointing Game metric. Overall, gradient-based interpretability approaches, such as Grad-CAM, Increment-CAM, and R-CAM, demonstrate relatively limited performance across all three metrics, particularly with noticeably lower IoU scores. This indicates that although these methods are able to roughly highlight infection-related regions, the resulting saliency

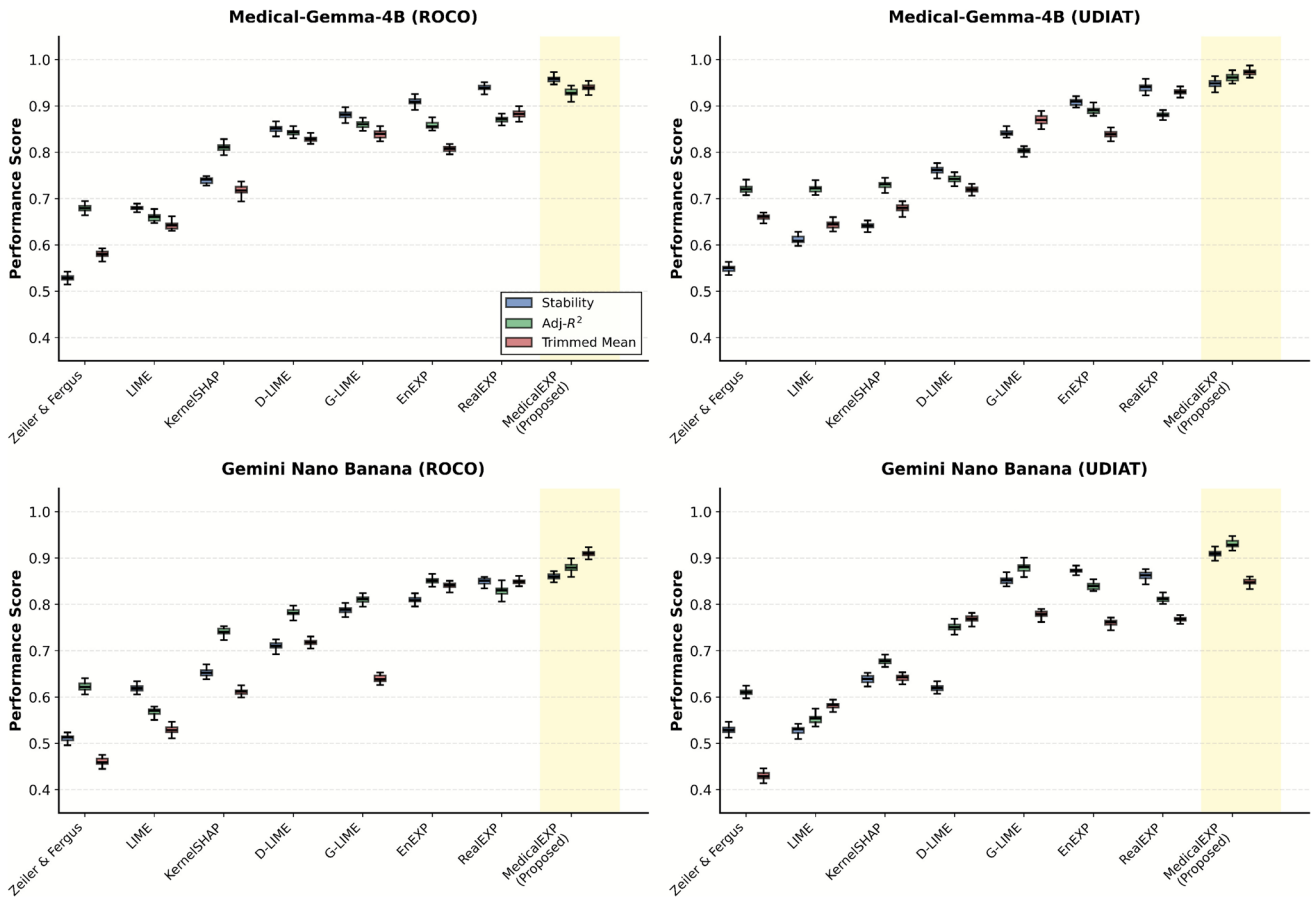


Fig. 5 Standard deviation experiment in different perturb-based interpretable machine learning methods

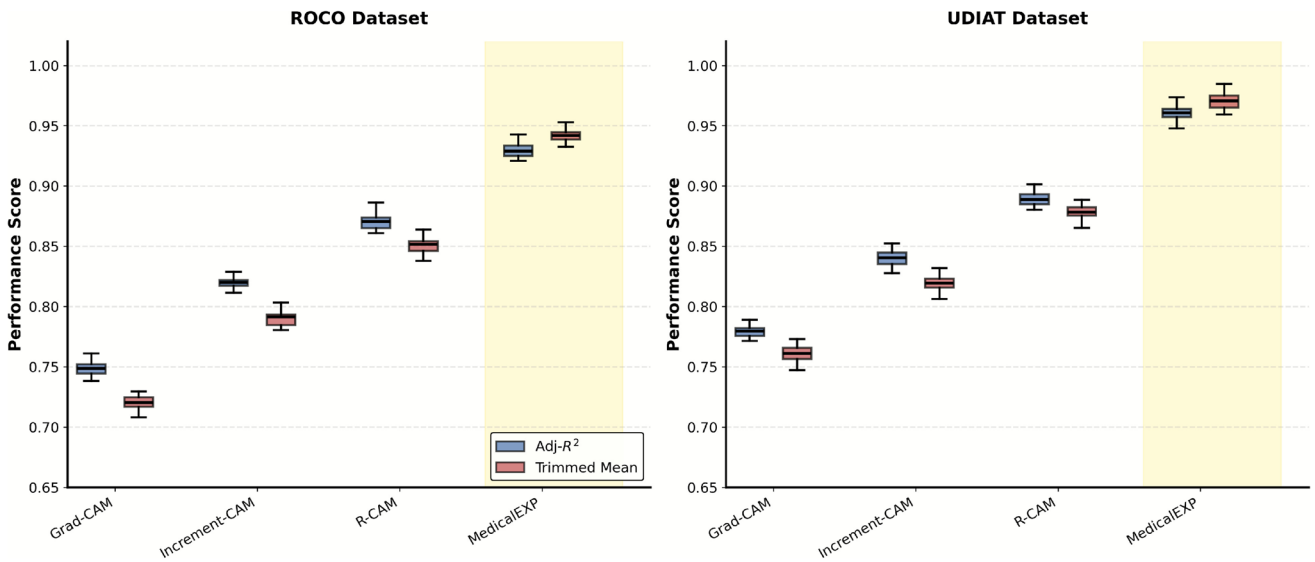


Fig. 6 Standard deviation experiment in different gradient-based interpretable frameworks

Table 4 Comparative evaluation of interpretability methods on the COVID-19 CT Scans dataset using ROI-based metrics

Method	Coverage	IoU	PointingGame
COVID-19 CT Scans Dataset			
Grad-CAM (Selvaraju et al. 2017)	0.48	0.36	0.52
Increment-CAM	0.52	0.39	0.56
R-CAM	0.55	0.41	0.59
(Zeiler and Fergus 2014)	0.50	0.38	0.54
LIME (Ribeiro et al. 2016)	0.58	0.44	0.61
KernelSHAP (Lundberg and Lee 2017)	0.61	0.47	0.64
D-LIME (Zafar and Khan 2021)	0.66	0.52	0.69
G-LIME (Tan et al. 2024)	0.72	0.58	0.75
EnEXP (Lee et al. 2025)	0.75	0.61	0.78
RealEXP (Jiang et al. 2025)	0.78	0.64	0.82
LFP (Chih et al., 2025)	0.76	0.62	0.80
Proposed MedExplainer	0.85	0.72	0.90

maps tend to be spatially diffuse and poorly aligned with the true lesion areas. Traditional perturbation-based methods, including Zeiler & Fergus, LIME, and KernelSHAP, achieve moderate improvements in ROI coverage and localization accuracy; however, they still struggle to provide complete and consistent coverage of complex infection patterns in CT images.

In contrast, more recent enhanced and ensemble-based perturbation methods, such as D-LIME, G-LIME, EnEXP, RealEXP, and LFP, show clear advantages across all three evaluation metrics, suggesting that structured perturbation strategies and ensemble mechanisms effectively improve spatial alignment between explanations and ground-truth lesions. Among these methods, RealEXP and LFP achieve relatively strong performance in terms of Coverage, IoU, and Pointing Game accuracy, indicating robust localization capability. Notably, the proposed MedExplainer consistently outperforms all competing approaches, achieving an ROI Coverage of 0.85, an IoU of 0.72, and a Pointing Game accuracy of 0.90. These results demonstrate that the proposed method not only provides more comprehensive coverage of infection regions but also achieves superior spatial alignment and reliable key-point localization, further validating its effectiveness and robustness for interpretable analysis of medical CT imaging.

Figure 7 illustrates substantial differences in computational complexity among various perturbation-based interpretability methods. As the number of perturbed samples increases from 50 to 200, all methods exhibit a roughly linear growth in average runtime; however, the slope and overall computational cost differ considerably. For example, the runtime of KernelSHAP increases from approximately

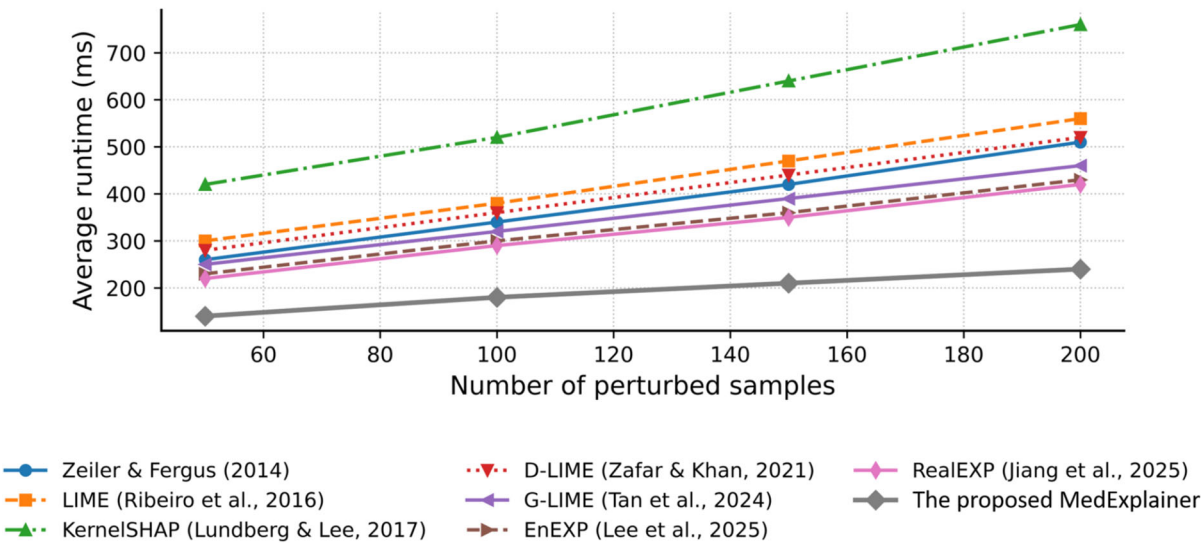


Fig. 7 Complexity comparison of perturbation-based interpretable frameworks

420 ms to 760 ms, representing a rise of more than 80 its relatively high computational burden. LIME and D-LIME fall within the 300–560 ms range, while G-LIME, EnEXP, and [RealEXP remain between 230 and 450 ms, reflecting moderate complexity. In comparison, the method of Zeiler and Fergus operates within the 250–500 ms interval, showing behavior similar to other mid-level approaches.

Notably, the proposed MedExplainer achieves the lowest computational complexity among all evaluated methods. Under identical perturbation settings, its average runtime increases only from approximately 150 ms to 240 ms, with a total increase of less than 60%. In addition to having the lowest absolute runtime, its growth rate is significantly slower than that of competing methods. When the number of perturbed samples reaches 200, MedExplainer operates at roughly one-third the runtime of KernelSHAP and half that of LIME, demonstrating superior efficiency and stability even under large-scale perturbation conditions. These findings indicate that MedExplainer not only provides stable interpretability but also offers a clear computational advantage over existing perturbation-based frameworks, making it particularly well suited for high-throughput medical imaging applications.

Figure 8 compares the average runtime of several gradient-based interpretable methods –Grad-CAM, Increment-CAM, R-CAM, and CR-CAM—with the proposed MedExplainer under the same hardware configuration and model architecture. As shown in the figure, existing gradient-based methods exhibit relatively high computational costs, with average runtimes ranging from 210 to 260 ms. Among them, CR-CAM is the most time-consuming (approximately 260 ms), while R-CAM and Increment-CAM require around 230 ms and 240 ms, respectively. Even the classical Grad-CAM takes about 210 ms, indicating that multilayer gradient backpropagation and feature-map weighting still impose notable computational overhead.

In contrast, the proposed MedExplainer demonstrates a substantial efficiency advantage, achieving an average runtime of only 160 ms, which is approximately 23.8% faster than Grad-CAM, 33.3% faster than Increment-CAM, and 38.5% faster than CR-CAM. This improvement primarily arises from MedExplainer's lightweight structured perturbation strategy and segmented inference mechanism, allowing the method to estimate regional importance without relying on deep gradient backpropagation. Therefore, in terms of computational complexity, MedExplainer not only reduces processing cost significantly but is also more suitable for real-time or high-throughput medical image interpretation scenarios.

Figure 9 presents a qualitative comparison of different explanation methods on two medical imaging benchmarks: (a) the ROCO dataset and (b) the UDIAT dataset, highlighting explanation stability and fidelity under repeated perturbations. For each dataset, the leftmost column shows the original input image and the corresponding ground-truth lesion mask, while the remaining columns visualize the saliency maps produced by Zeiler & Fergus, LIME, D-LIME, G-LIME, EnEXP, RealEXP, and the proposed MedExplainer (Ours) across multiple perturbation instances.

From the perspective of explanation stability, both subfigures indicate that traditional perturbation-based methods (e.g., Zeiler & Fergus, LIME, and D-LIME) generate saliency maps with scattered and inconsistent activations, where the highlighted regions vary substantially in location, shape, and intensity across perturbations, reflecting strong sensitivity to randomness and limited robustness. More recent methods such as G-LIME, EnEXP, and RealEXP partially improve spatial coherence; however, their explanations still exhibit noticeable diffusion, spatial shifts, or fragmented activations, especially in complex anatomical structures. In contrast, MedExplainer produces compact and spatially consistent

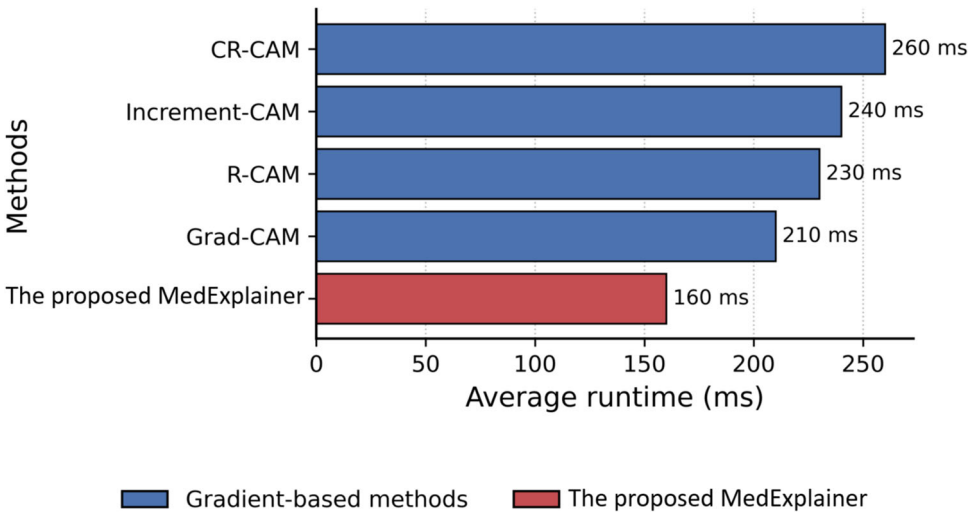
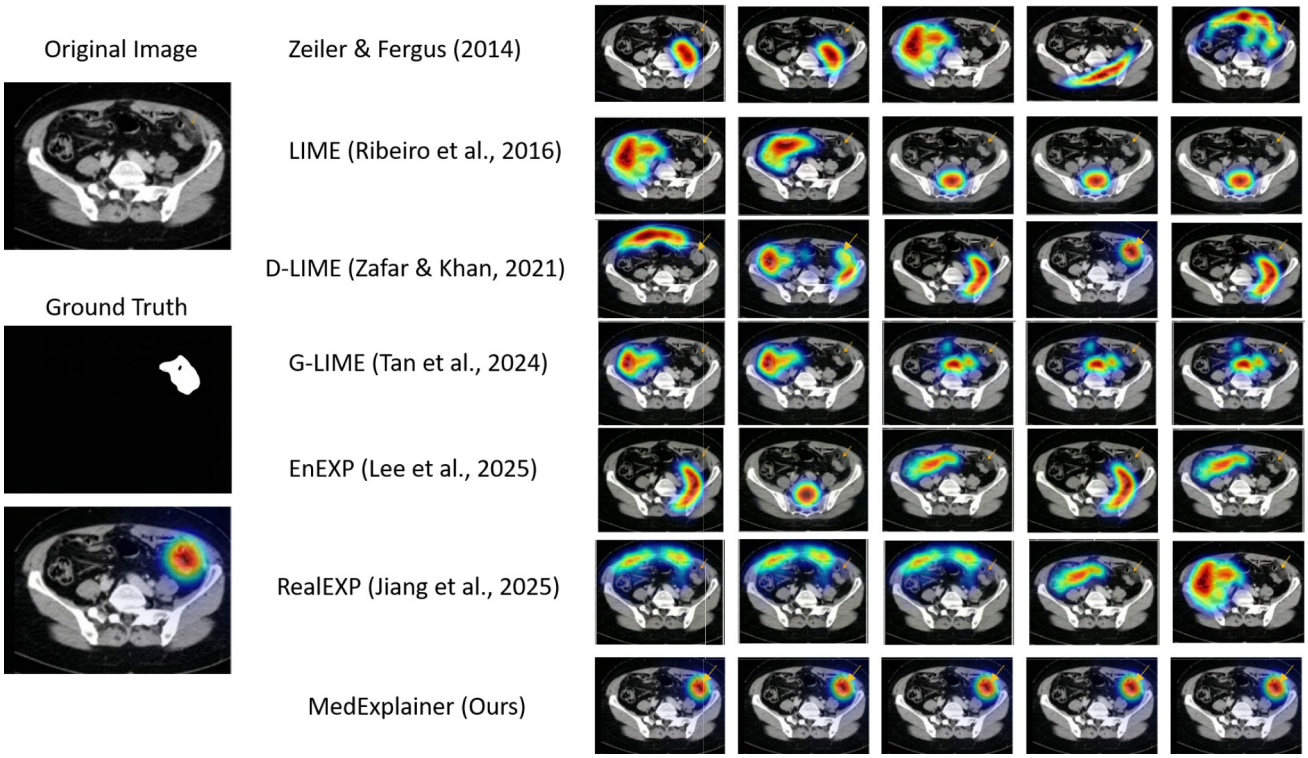
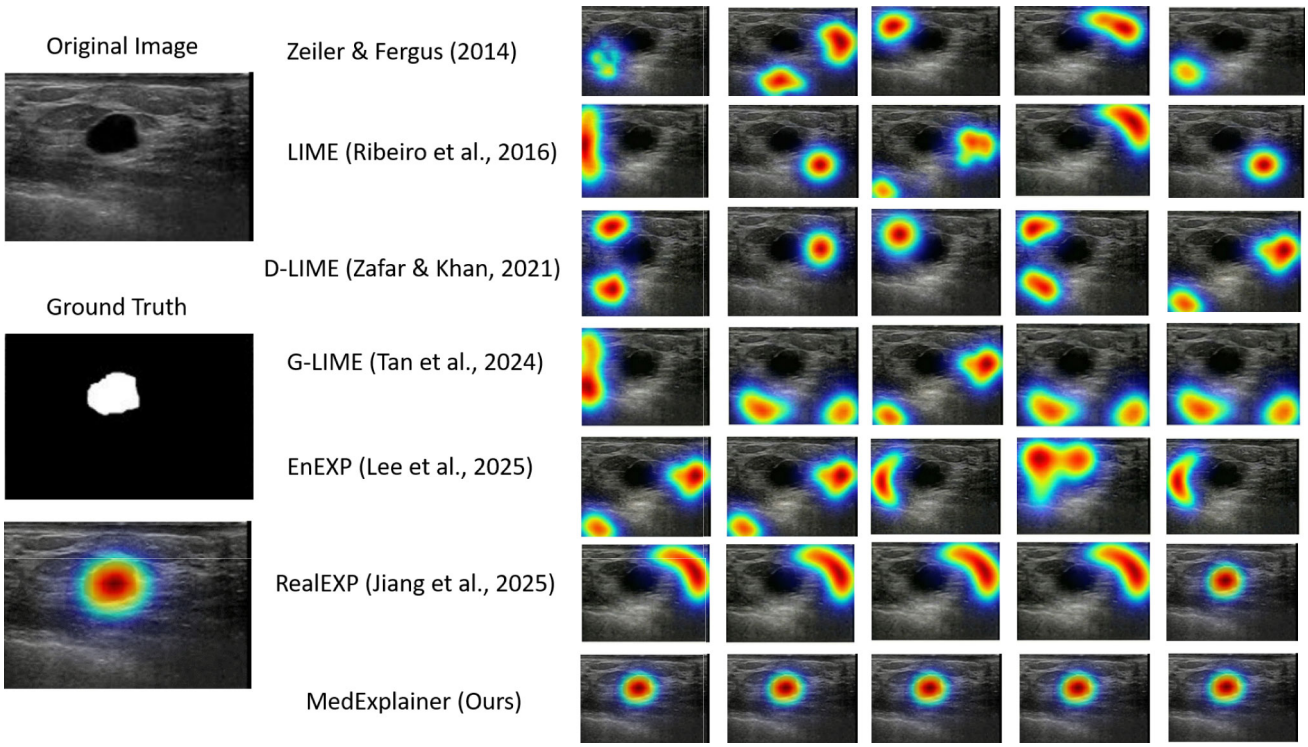


Fig. 8 Runtime comparison of gradient-based interpretable methods



(a) ROCO dataset



(b) UDIAT dataset

Fig. 9 Qualitative comparison highlighting the stability and fidelity of the proposed MedExplainer

activation regions across perturbations in both ROCO and UDIAT, demonstrating superior stability.

From the perspective of explanation fidelity, Fig. 9 further shows that MedExplainer achieves markedly better spatial alignment with the ground-truth lesion regions. As illustrated in (a), MedExplainer accurately highlights the clinically relevant regions while suppressing irrelevant background responses, and in (b) it precisely localizes lesion areas in ultrasound images despite speckle noise and low contrast. In comparison, competing methods often activate non-diagnostic background regions or overly extended areas beyond the true lesion boundaries. Overall, these qualitative results demonstrate that MedExplainer provides explanations that are both stable under perturbations and faithful to true pathological regions, consistent with the quantitative evaluations reported earlier.

4.5 Ablation study

This section will introduce the ablation experiments of the designed MedExplainer to demonstrate the effectiveness of each module.

From the ablation results reported in Table 5, it can be clearly observed that each key component of the proposed method plays a significant role in improving interpretability performance, and consistent performance gains are observed across different datasets. On the UDIAT dataset, when only Kernel Adjust and Parallel Trees are enabled without Fixed Perturb, the model achieves a Stability of 0.54, an $AdjR^2$ of 0.62, and a Trimmed Mean of 0.67, indicating relatively limited performance. After introducing Fixed Perturb, the Stability increases to 0.62, the $AdjR^2$ to 0.68, and the Trimmed Mean to 0.77, corresponding to absolute improvements of approximately 0.08, 0.06, and 0.10, respectively. These results indicate that enforcing a fixed perturbation ratio effectively reduces randomness-induced variability and directly enhances explanation consistency.

Further analysis shows that enabling Fixed Perturb and Parallel Trees without Kernel Adjust leads to noticeable performance improvements but still fails to achieve optimal results. On the ROCO dataset, this configuration yields Stability, $AdjR^2$, and Trimmed Mean values of 0.76, 0.73, and 0.82, representing increases of 0.12, 0.06, and 0.19 over the configuration without Fixed Perturb. However, when Kernel Adjust is additionally incorporated, all three metrics improve simultaneously to 0.96, 0.96, and 0.94, with Stability and $AdjR^2$ increasing by 0.20 and 0.23, respectively. This observation demonstrates that similarity-based weighting plays a crucial role in enhancing surrogate model fitting and explanation fidelity, and that ensemble structures alone are insufficient to accurately model the relative importance of different perturbation samples.

The ablation results on the COVID-19 CT Scans dataset further support these findings. When only partial components are enabled, the model exhibits limited performance on ROI-based metrics. For instance, without Fixed Perturb, the Coverage, IoU, and Pointing Game scores are 0.56, 0.41, and 0.63, respectively. After introducing Fixed Perturb, these values increase to 0.68, 0.53, and 0.74, corresponding to improvements of approximately 0.12, 0.12, and 0.11. Notably, only when all three components—Fixed Perturb, Kernel Adjust, and Parallel Trees—are jointly enabled does the model achieve optimal performance, with Coverage reaching 0.85, IoU reaching 0.72, and Pointing Game reaching 0.90. Compared with partial configurations, the IoU and Pointing Game scores improve by approximately 0.19 and 0.16, respectively. Taken together, the ablation results across all three datasets demonstrate that the proposed components

Table 5 Ablation study of the proposed method on the UDIAT, ROCO, and COVID-19 CT Scans datasets. The VLM used in this experiment is Medical-Gemma-4B with LoRA adaptation

Fixed Perturb	Kernel Adjust	Parallel Trees	Stability	$AdjR^2$	Trimmed Mean
UDIAT Dataset					
	✓	✓	0.54	0.62	0.67
✓		✓	0.62	0.68	0.77
✓	✓	✓	0.85	0.96	0.97
ROCO Dataset					
	✓	✓	0.64	0.67	0.63
✓		✓	0.76	0.73	0.82
✓	✓	✓	0.96	0.96	0.94
COVID-19 CT Scans Dataset					
Fixed Perturb	Kernel Adjust	Parallel Trees	Coverage	IoU	PointingGame
	✓	✓	0.56	0.41	0.63
✓		✓	0.68	0.53	0.74
✓	✓	✓	0.85	0.72	0.90

exhibit strong complementarity in terms of explanation stability, spatial consistency, and localization accuracy, and their joint effect is critical for achieving optimal interpretability performance as shown in Fig. 10.

Across all experiments, including both Medical–Gemma–4B and Gemini Nano Banana, the three evaluation metrics—Stability, AdjR^2, and Trimmed Mean—exhibit a consistent pattern as the hyperparameter λ in Eq. (14) varies. When λ is too small (e.g., 0.1 or 0.2), the performance of all metrics remains relatively low, with Stability showing the most noticeable degradation. This indicates that weak weighting adjustments fail to properly integrate perturbation patterns, resulting in unstable explanations. Conversely, when λ is too large (e.g., 0.4 or 0.5), performance slightly declines again, suggesting that overemphasizing perturbation-based weighting undermines the robustness of the global structure.

At $\lambda = 0.3$, both Medical–Gemma–4B and Gemini Nano Banana achieve the highest performance across all metrics. Stability approaches or reaches 1.0, demonstrating that explanations remain highly consistent across perturbation samples. AdjR^2 and Trimmed Mean also reach their peak values, indicating that the inferred feature-importance rankings strongly correspond to model output variations while maintaining stable aggregated behavior. As illustrated in the figure, the surface plot at $\lambda = 0.3$ displays smoother curvature, minimal fluctuations, and the highest peak values, confirming that this setting provides the optimal balance between perturbation weighting and global structural integrity.

A similar trend is consistently observed across both a large-scale VLM (Medical–Gemma–4B) and a lightweight model (Gemini Nano Banana). Both models achieve their best results at $\lambda = 0.3$, showing that this optimal hyperparameter setting does not depend on any specific architecture or scale. This cross-model consistency indicates that the tuning mechanism introduced in (14) is generalizable and stable across different medical vision–language models, making $\lambda = 0.3$ a robust choice for maintaining interpretability stability and overall explanation quality.

Figure 11 presents the ablation study on the effect of the number of perturbation samples across three datasets, including UDIAT, ROCO, and COVID-19 CT Scans. It can be observed that, when the number of perturbations is relatively small (ranging from 50 to 800), all evaluation metrics exhibit only marginal improvements with minor fluctuations. This indicates that insufficient perturbation diversity limits the model's ability to fully capture stable and discriminative representations. In contrast, a substantial performance gain is consistently observed when the perturbation number reaches 1000, suggesting that a sufficient number of perturbations is critical for effective model optimization.

From the perspective of individual metrics, similar trends can be found across different datasets. Metrics related to prediction stability and localization quality show gradual growth under low perturbation settings, implying limited robustness against input variations. As the perturbation number increases, especially at the highest setting, these metrics improve significantly, demonstrating that extensive perturbation sampling helps the model learn more reliable and consistent responses. Notably, the performance gap between low and high perturbation regimes highlights the importance of adequately covering the perturbation space during training or evaluation.

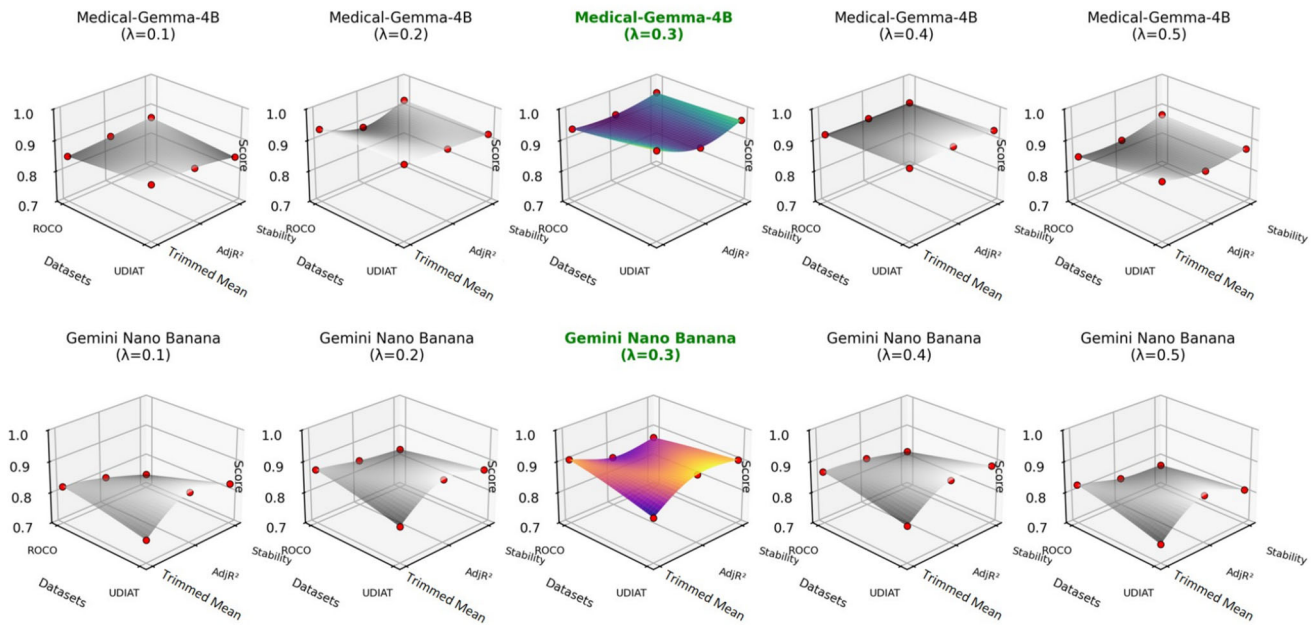


Fig. 10 Discussion of hyperparameters under different settings

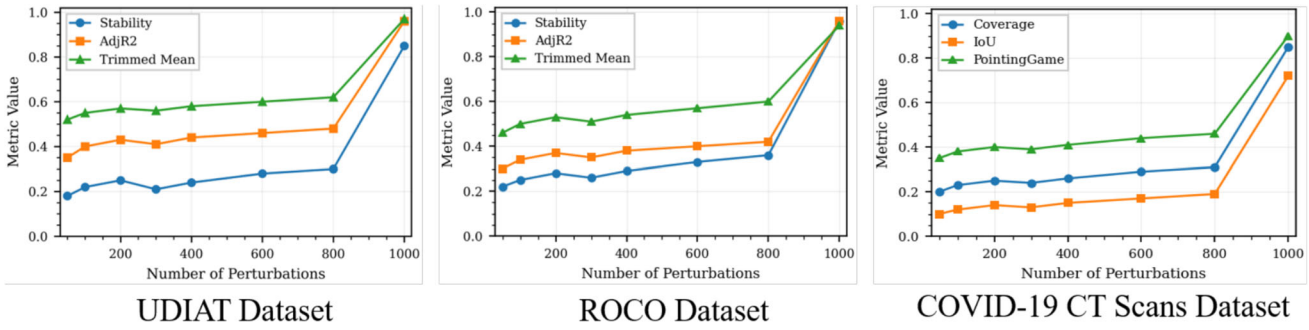


Fig. 11 Discussion of perturb samples under different numbers

Although the three datasets differ substantially in imaging modality, data distribution, and task complexity, they all present highly consistent performance patterns with respect to perturbation numbers. This consistency suggests that the observed improvements are not dataset-specific, but rather reflect a general property of the proposed method. Based on these observations, a perturbation number of 1000 is adopted in subsequent experiments, as it provides a favorable balance between performance gains and computational cost.

Figure 12 illustrates the sensitivity of model performance to the number of trees and tree depth across three datasets. In this experiment, the x-axis denotes the tree depth, the y-axis represents the number of trees, and the z-axis corresponds to the performance metrics. Across all datasets, the performance surfaces exhibit a clear non-linear relationship with respect to both hyperparameters, indicating that neither excessively shallow nor overly deep trees are optimal. Instead, a moderate configuration yields the most favorable results.

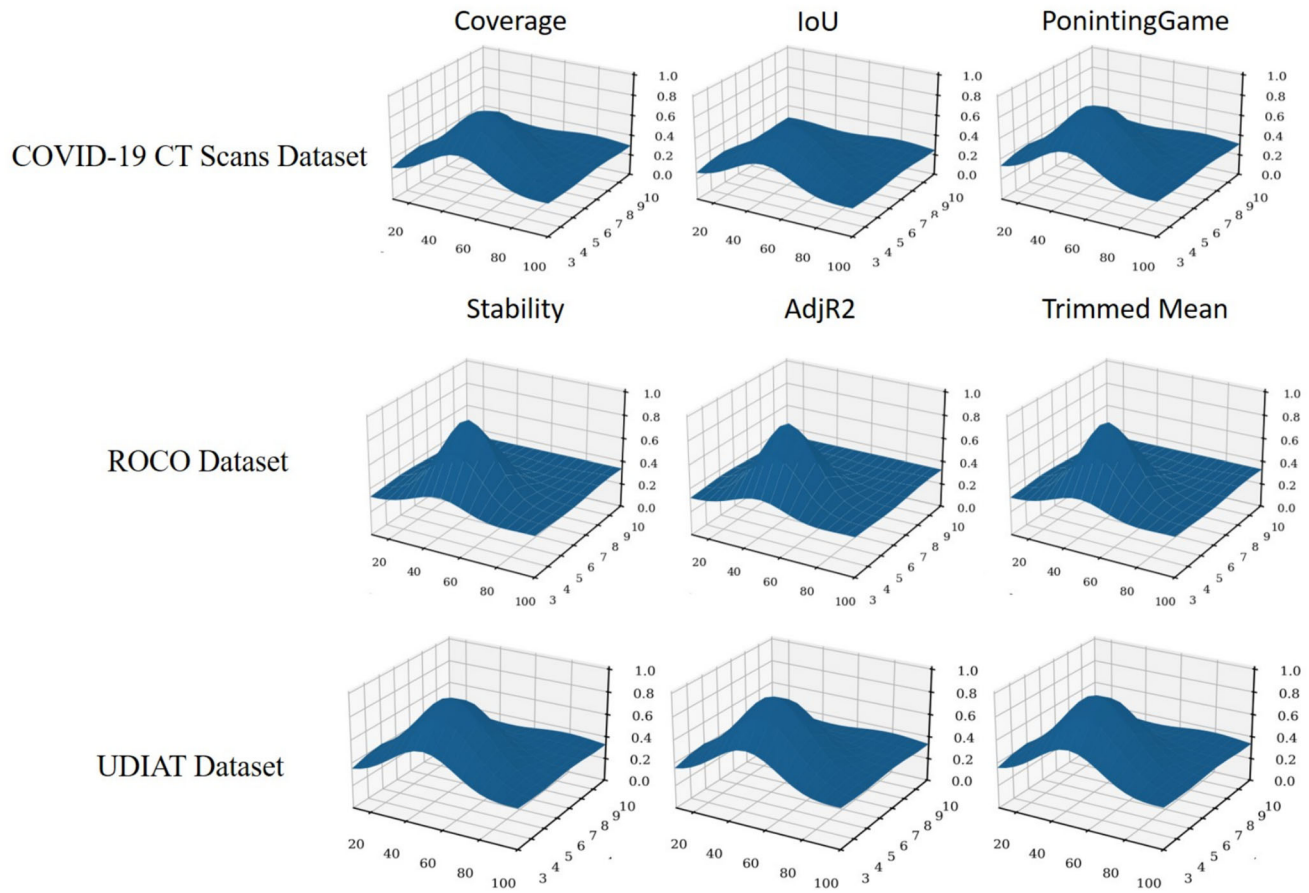


Fig. 12 Sensitivity analysis of tree number and depth on model performance

For the ROCO and UDIAT datasets, the performance peaks consistently emerge around a tree depth of 5 and a tree number close to 50. When the number of trees is small or the depth is insufficient, the model suffers from limited representational capacity, resulting in suboptimal performance. Conversely, increasing the tree depth or tree number beyond this range leads to marginal gains or even slight degradation, suggesting potential overfitting or redundancy in the ensemble structure. This trend is consistently reflected across all three evaluation metrics.

A similar pattern can be observed on the COVID-19 CT Scans dataset, although the performance landscape appears more irregular, reflecting the higher complexity and noise characteristics of medical CT images. Nevertheless, the optimal region remains concentrated around moderate tree depths and tree numbers. These consistent observations across different datasets demonstrate that a balanced combination of tree number and depth is critical for achieving robust and generalizable performance.

Figure 13 analyzes the model performance under different *mask ratios* on ROCO, UDIAT, and COVID-19 CT Scans datasets. The curves consistently exhibit a clear unimodal trend with respect to the mask ratio: the performance first improves as the mask ratio decreases from 1.0, and then drops rapidly when the ratio approaches 0. This observation indicates that neither overly strong masking nor the absence of masking is optimal for the proposed method.

For ROCO and UDIAT, the best performance is achieved at an intermediate mask ratio (around 0.3). With large mask ratios, excessive occlusion disrupts semantic structures and undermines representation learning, leading to reduced stability and inferior fitting quality. In contrast, when the mask ratio becomes too small, the perturbation is insufficient to introduce meaningful variations, which limits the regularization effect and yields weaker improvements.

On the COVID-19 CT Scans dataset, a similar pattern is observed while the performance landscape appears more irregular, reflecting the higher noise level and complexity of CT images. Among the three metrics, PointingGame remains relatively robust across a broader range of mask ratios, whereas IoU is more sensitive to strong masking. Nevertheless, the optimal region is still concentrated around the intermediate ratio, demonstrating that a properly calibrated mask ratio provides a favorable balance between semantic preservation and perturbation diversity.

Overall, the consistent peak around a mask ratio of 0.3 across datasets and metrics highlights the importance of calibrating masking strength. These results suggest that an appropriate mask ratio can effectively enhance robustness without sacrificing semantic integrity. Therefore, we adopt a mask ratio of 0.3 as the default setting in subsequent experiments.

Figure 14 compares the robustness of ensemble models trained with shared perturbations and diverse perturbations under increasing out-of-distribution (OOD) severity across three datasets. As the OOD severity increases, the performance of both training strategies degrades consistently, reflecting the increasing difficulty caused by distributional shifts. Nevertheless, the models trained with diverse perturbations consistently maintain higher performance across all metrics and datasets, indicating a clear advantage in robustness.

On the ROCO and UDIAT datasets, the performance gap between the two strategies becomes increasingly pronounced as the OOD severity grows. While both approaches exhibit comparable performance under mild distribution shifts, the shared-perturbation strategy suffers from a noticeably faster decline as the shift becomes stronger, particularly in terms of Stability and AdjR2. In contrast, the diverse-perturbation strategy demonstrates a more gradual degradation, suggesting that enhanced ensemble diversity effectively alleviates the negative impact of distribution mismatch.

A similar trend is observed on the COVID-19 CT Scans dataset, where performance decreases more rapidly due to the higher noise level and structural variability inherent in CT imaging. Despite this challenge, the diverse-perturbation model consistently outperforms its shared-perturbation counterpart across Coverage, IoU, and PointingGame metrics. These results confirm that introducing diversity through independent perturbations significantly

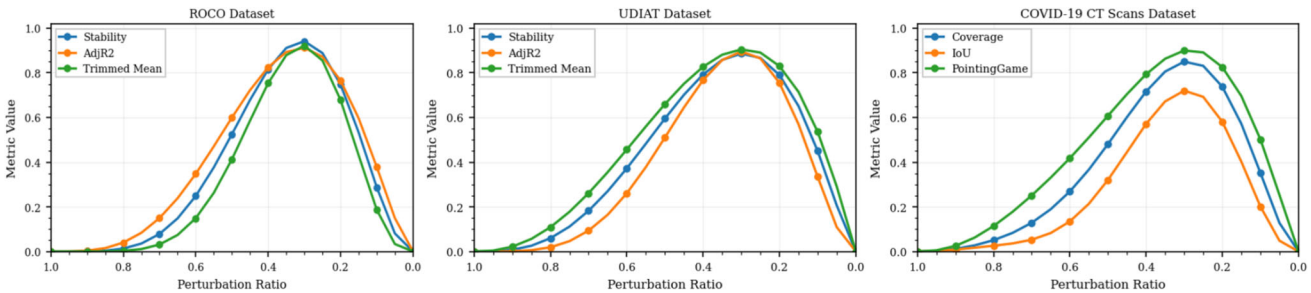


Fig. 13 Exploring the model performance under different mask ratios

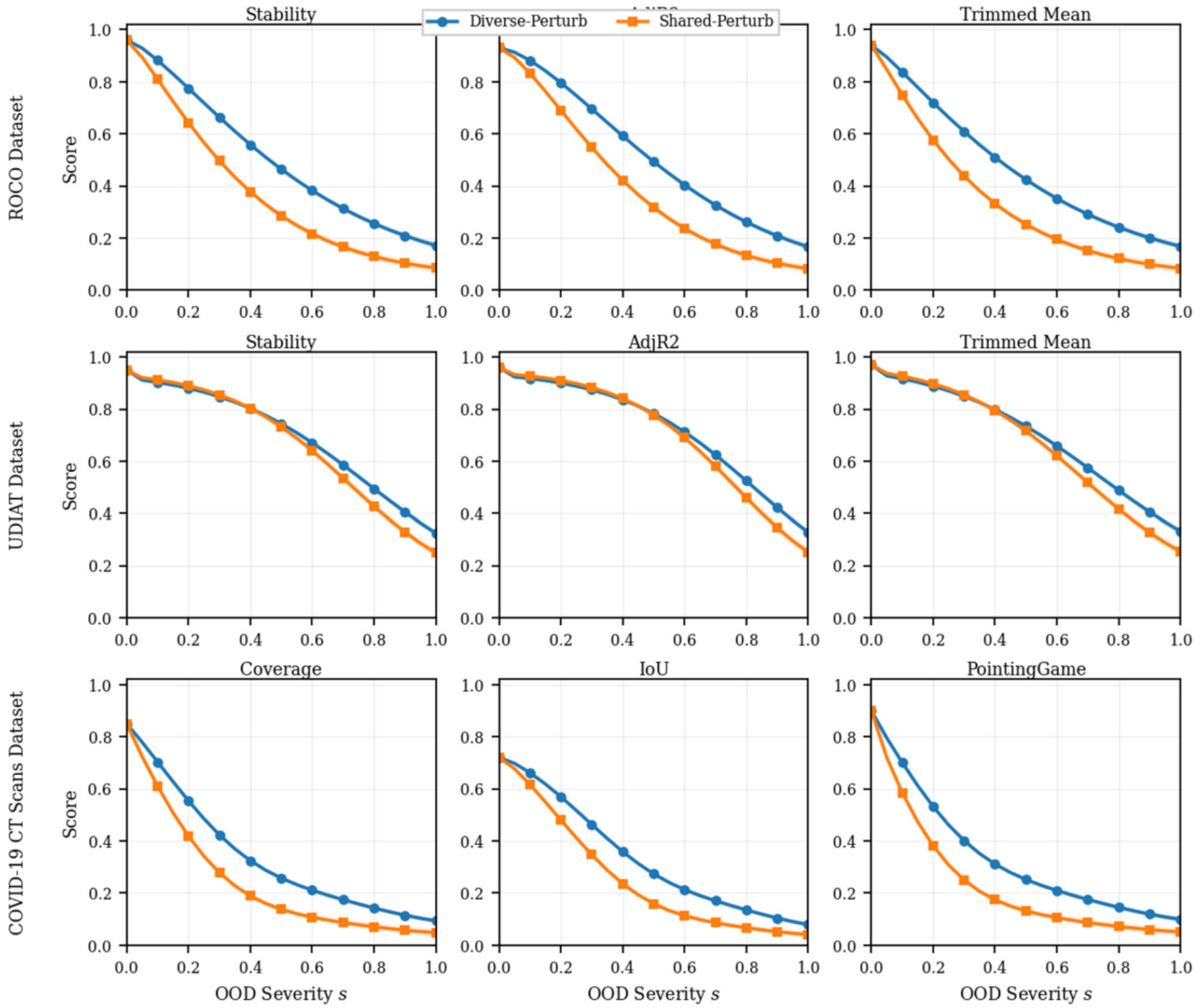


Fig. 14 Robustness under OOD shifts: shared vs diverse perturbations

improves robustness to OOD shifts, directly addressing concerns regarding limited ensemble diversity when all trees are trained on identical perturbed data.

As shown in Fig. 15(a)(b), surrogate models exhibit clear performance differences on the ROCO and UDIAT medical image datasets in terms of Stability, Adjusted R^2 (Adj-R^2), and Trimmed Mean. The linear model (LIME) consistently shows the weakest performance, while Random Forest and XGBoost benefit from their nonlinear modeling capacity and achieve moderate improvements. In contrast, the proposed Parallel-Tree Ensemble achieves the best results across both datasets. In particular, it consistently attains values above 0.95 in both Stability and Adj-R^2, indicating that it can produce highly consistent and well-fitted local explanations under a fixed perturbation strategy.

For the CT lesion localization task (Fig. 15(c)), the Parallel-Tree Ensemble also outperforms all baseline methods across Coverage, IoU, and Pointing Game metrics. Compared with LIME and Random Forest, it improves Coverage by approximately 20% and increases IoU by more than 0.2. These results demonstrate that the proposed method not only achieves more accurate numerical fitting but also provides more reliable spatial localization of clinically relevant regions. This confirms its superior capability in capturing complex structures and locally discriminative features in high-dimensional medical images.

From the perspective of computational efficiency (Fig. 15(d)), the runtime of Random Forest and XGBoost increases significantly as the number of perturbations grows, whereas the proposed method exhibits an approximately linear scaling

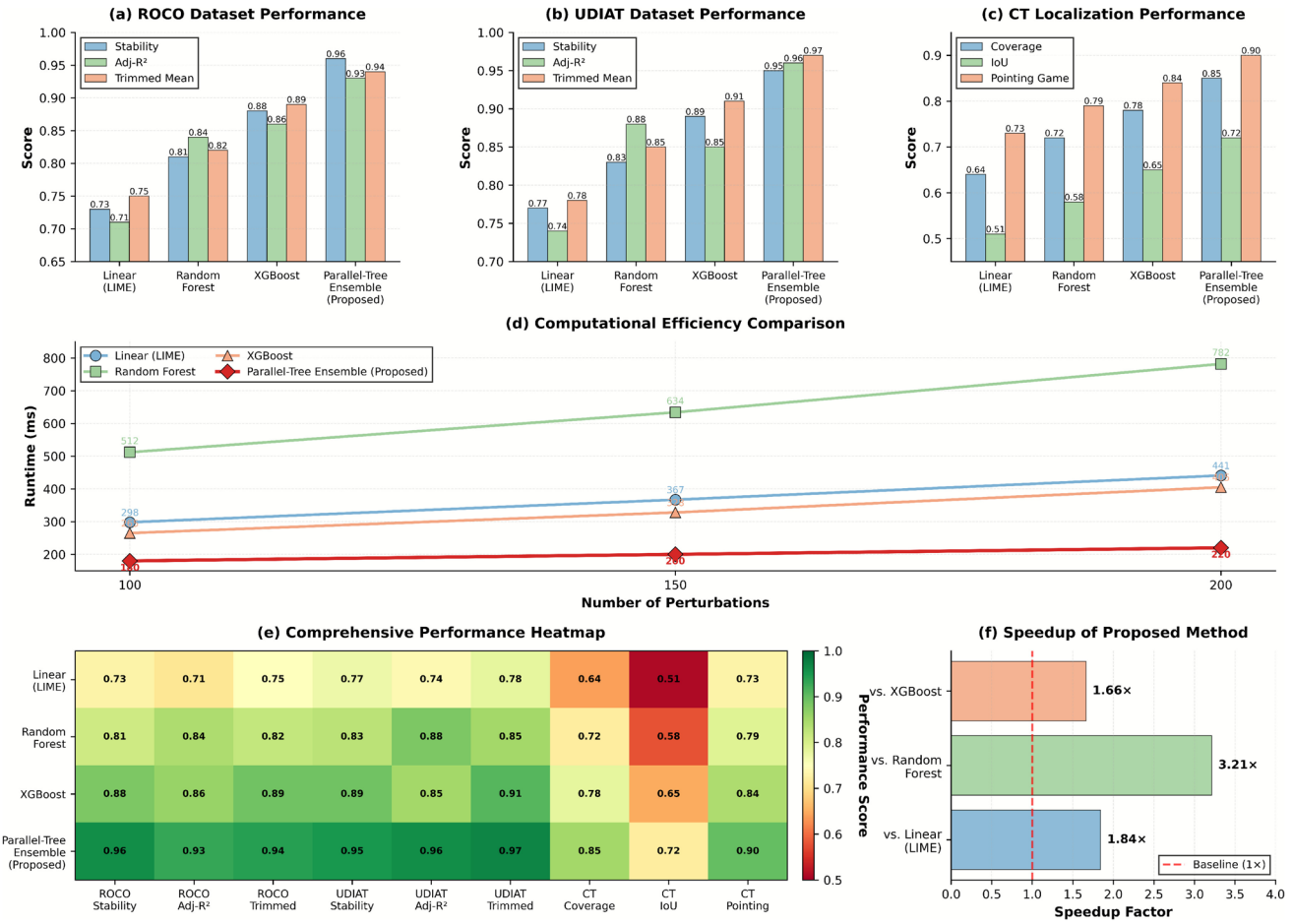


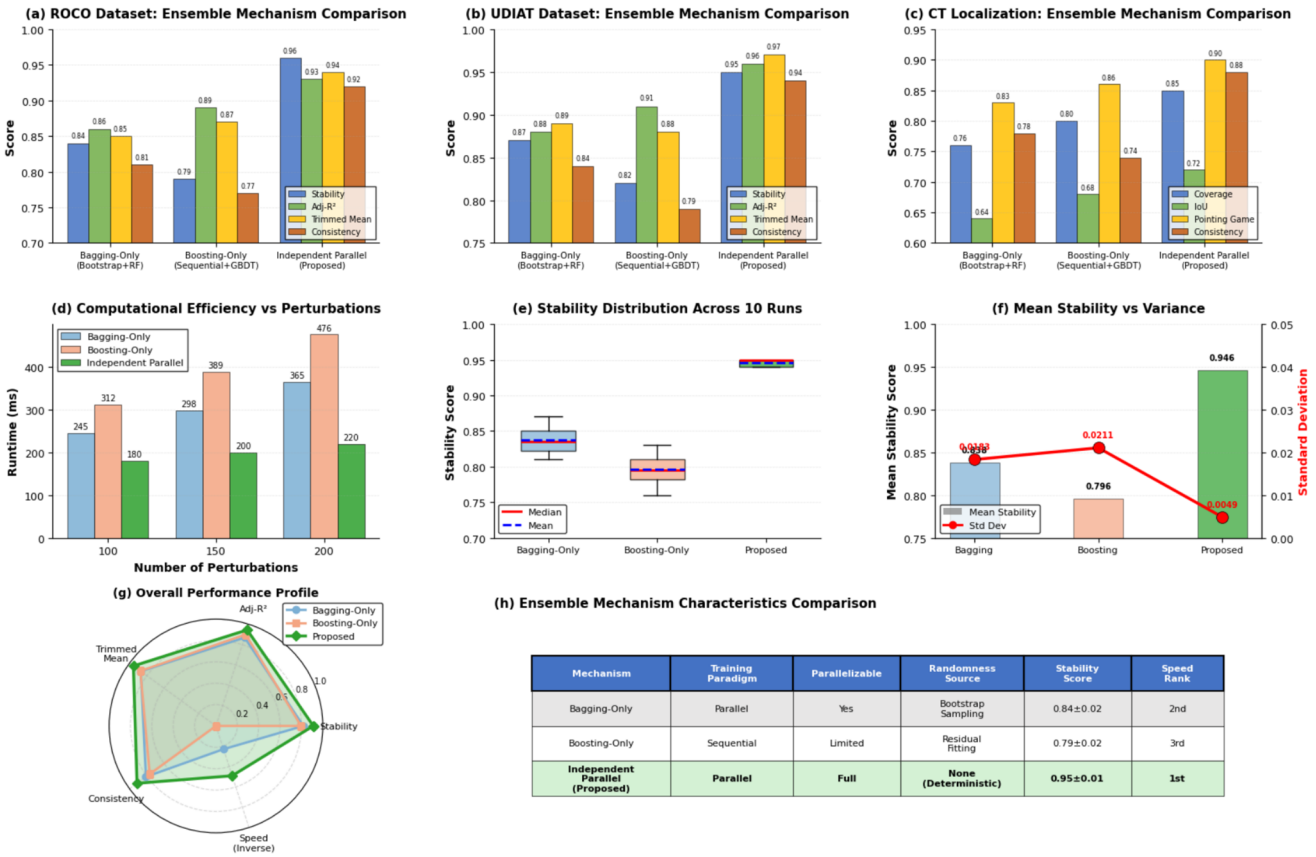
Fig. 15 Comprehensive comparison of surrogate models for medical image explanation on fixed perturbation strategy: superpixel-based, K Samples, Masking Ratio=0.3, Kernel Weighting=0.3

behavior and consistently maintains the lowest inference cost. At 200 perturbations, its runtime is only about 220 ms, substantially outperforming the competing approaches. Furthermore, the speedup analysis in Fig. 15(f) shows that the proposed method achieves acceleration factors of 1.84×, 1.66×, and 3.21× compared to LIME, XGBoost, and Random Forest, respectively, highlighting the efficiency advantage of the parallel ensemble design during explanation generation.

The comprehensive heatmap in Fig. 15(e) further summarizes the overall performance across multiple datasets and evaluation metrics. It can be observed that the Parallel-Tree Ensemble achieves the highest or near-highest scores in all nine evaluation dimensions, while also exhibiting more balanced and stable behavior. Overall, these experimental results demonstrate that under a fixed superpixel-based perturbation strategy with unified parameter settings, the proposed Parallel-Tree Ensemble surrogate model significantly outperforms existing mainstream methods in terms of explanation accuracy, stability, spatial consistency, and computational efficiency, indicating strong practical applicability.

Figure 16 aims to investigate the impact of different ensemble training mechanisms on explanation performance, stability, and computational efficiency under a fixed perturbation strategy. Specifically, three representative ensemble paradigms are compared: Bagging-only (Bootstrap + Random Forest), Boosting-only (Sequential + GBDT), and the proposed Independent Parallel Ensemble, while keeping the perturbation strategy and hyperparameters unchanged.

As shown in Fig. 16(a)(b), on both the ROCO and UDIAT datasets, the Independent Parallel ensemble consistently achieves the highest scores across Stability, Adj-R^2, Trimmed Mean, and Consistency. While Bagging-only benefits from bootstrap-induced diversity and achieves relatively stable performance, its fitting accuracy is limited. In contrast, Boosting-only improves Adj-R^2 through sequential residual fitting but suffers from degraded stability and consistency due to inherent error accumulation. The proposed mechanism effectively combines strong fitting capability with deterministic behavior, leading to superior and more balanced performance across all metrics.



Mechanism	Training Paradigm	Parallelizable	Randomness Source	Stability Score	Speed Rank
Bagging-Only	Parallel	Yes	Bootstrap Sampling	0.84±0.02	2nd
Boosting-Only	Sequential	Limited	Residual Fitting	0.79±0.02	3rd
Independent Parallel (Proposed)	**Parallel**	**Full**	**None (Deterministic)**	**0.95±0.01**	**1st**

Fig. 16 Ablation study: impact of ensemble training mechanism (Fixed Perturbation Strategy: Superpixel-based, Masking Ratio=0.3, Kernel Weighting=0.3

For the CT localization task (Fig. 16(c)), similar trends can be observed. The Independent Parallel ensemble outperforms both Bagging-only and Boosting-only in Coverage, IoU, and Pointing Game metrics, indicating more reliable spatial localization of clinically relevant regions. This suggests that eliminating sequential dependency while maintaining ensemble diversity enables the surrogate model to better capture localized discriminative patterns in high-dimensional medical images.

From the efficiency and robustness perspective, Fig. 16(d) shows that the proposed method consistently maintains the lowest runtime as the number of perturbations increases, exhibiting near-linear scalability. Moreover, the stability distribution over 10 independent runs (Fig. 16(e)) demonstrates that the proposed method achieves not only the highest median stability but also the smallest variance. This observation is further confirmed in Fig. 16(f), where the Independent Parallel ensemble attains the highest mean stability (0.946) with a substantially lower standard deviation (0.0049), highlighting its robustness and reproducibility.

Finally, the radar chart in Fig. 16(g) and the summary table in Fig. 16(h) provide a comprehensive comparison of ensemble characteristics. The proposed mechanism achieves the best overall trade-off among stability, fitting accuracy, consistency, and speed, ranking first in stability and efficiency. Overall, these results demonstrate that the Independent Parallel training mechanism plays a critical role in achieving stable, accurate, and efficient medical image explanations, beyond what can be obtained by conventional bagging or boosting strategies alone.

Figure 17 aims to analyze the robustness of explanation consistency under increasing label noise and to evaluate the effectiveness of the proposed Trimmed Mean aggregation strategy. Specifically, the Global Consistency Index is measured as the label noise ratio increases, comparing the standard Aggregate Mean baseline with the proposed Trimmed Mean approach.

As shown in Fig. 17, the Aggregate Mean baseline exhibits severe degradation as label noise increases. When the noise ratio reaches 30%, the Global Consistency Index drops sharply to around 0.60, and further declines to approximately 0.32 at 50% noise, corresponding to a relative degradation of more than 65%. This behavior indicates that conventional mean aggregation is highly sensitive to noisy or corrupted samples, which can dominate the explanation outcome and significantly distort global consistency.

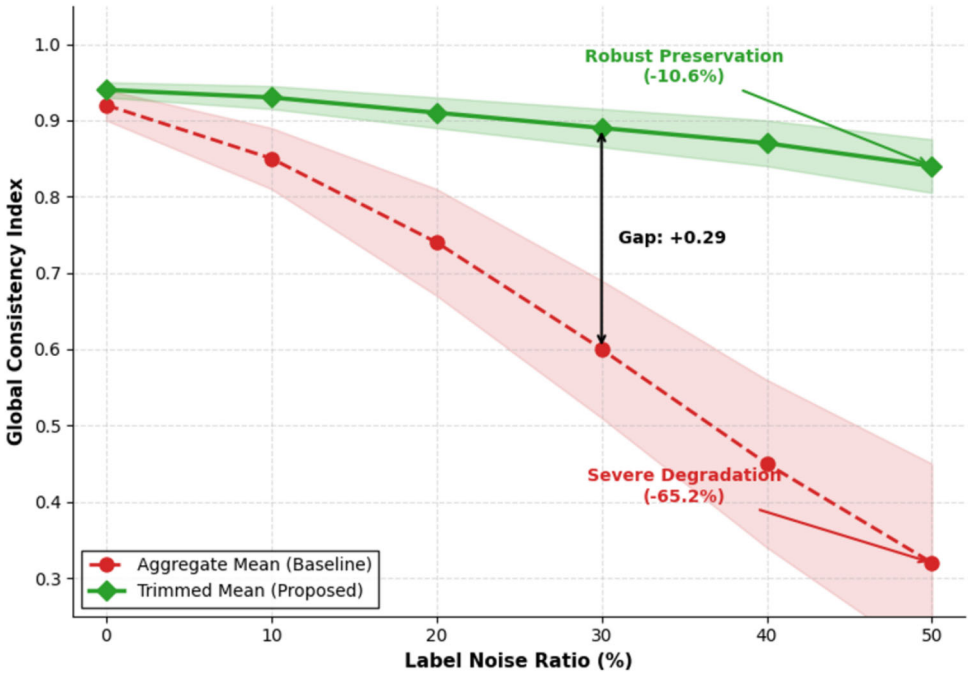


Fig. 17 Noise stress test analysis: trimmed mean effectiveness

In contrast, the proposed Trimmed Mean demonstrates robust preservation of explanation consistency across all noise levels. Even at 50% label noise, the consistency score remains above 0.85, with only a marginal degradation of approximately 10.6%. The growing performance gap between the two methods, reaching approximately 0.29 at 30% noise, clearly highlights the ability of the Trimmed Mean to effectively suppress the influence of extreme or unreliable perturbation samples.

Overall, these results confirm that the Trimmed Mean aggregation mechanism substantially enhances the robustness of explanation generation under noisy supervision. By mitigating the impact of outliers induced by label noise, the proposed strategy ensures more stable and reliable global explanations, which is particularly critical for real-world medical imaging scenarios where annotation noise is often unavoidable.

Figure 18 aims to analyze the sensitivity of explanation performance to superpixel granularity and to quantify the impact of superpixel boundary failures on localization and stability. This experiment evaluates how variations in superpixel segmentation quality affect both explanation accuracy and robustness under a fixed perturbation strategy.

As shown in Fig. 18(a), localization accuracy initially improves as superpixel granularity increases, reaching an optimal range around 100–150 superpixels, which is highlighted as the configuration used in the main experiments. Beyond this range, further increasing granularity leads to a gradual decline in localization performance, likely due to fragmented regions that disrupt semantic coherence. In contrast, explanation stability consistently decreases as granularity increases, indicating that overly fine superpixel partitions amplify sensitivity to local perturbations and reduce explanation robustness. This trade-off demonstrates the necessity of carefully balancing spatial resolution and stability when selecting superpixel parameters.

Figure 18(b) further quantifies the impact of superpixel boundary failure cases. When superpixel segmentation successfully preserves object boundaries, both localization and stability remain high. However, in failure cases with low correspondence to semantic structures, localization accuracy drops by approximately 47.2%, while stability decreases by about 27.3%. These results indicate that boundary leakage and region misalignment can significantly distort explanation quality, even when the underlying surrogate model remains unchanged.

Overall, the results in Fig. 18 highlight that superpixel quality plays a critical role in explanation reliability. While the proposed explanation framework remains robust within a reasonable granularity range, severe segmentation failures can still propagate errors into the explanation process. This analysis underscores the importance of selecting appropriate superpixel configurations and motivates future integration of adaptive or semantics-aware segmentation strategies to further enhance robustness.

Figure 19 aims to investigate the sensitivity of explanation performance to the masking ratio ρ and to identify a robust operating region that balances stability, fidelity, and localization accuracy. This experiment systematically varies the masking ratio while evaluating Stability, Fidelity (Adj-R^2), and Localization (IoU) under a fixed perturbation and surrogate modeling setting.

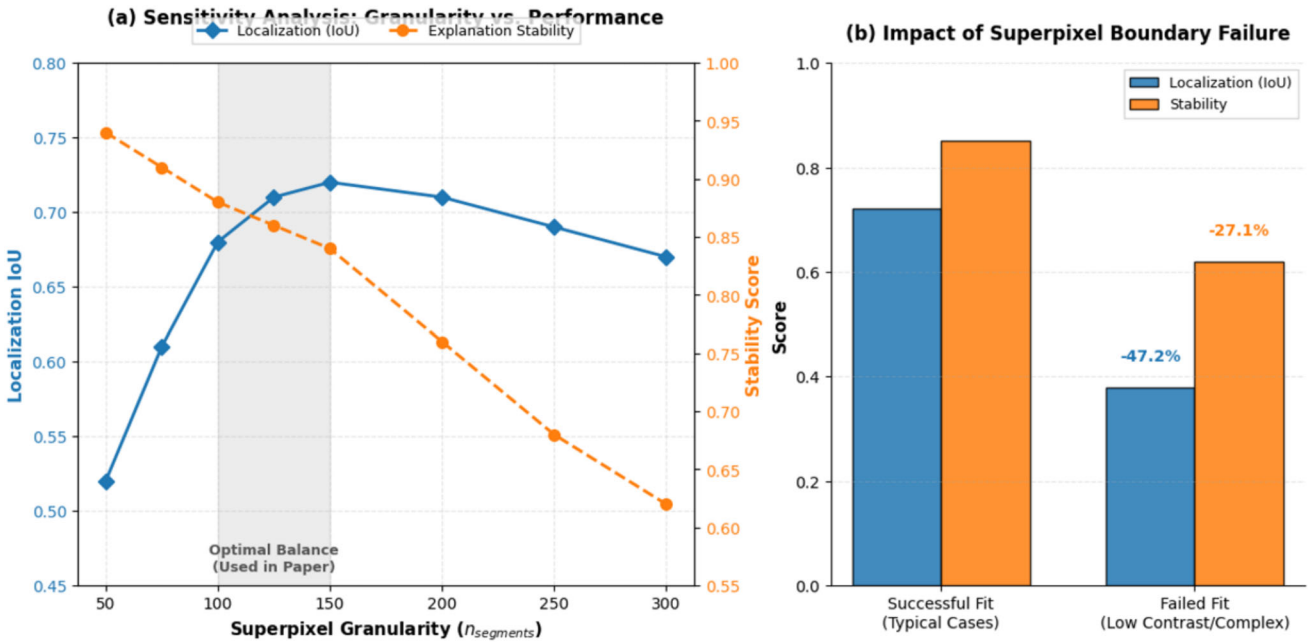


Fig. 18 Superpixel: sensitivity analysis and failure case quantification

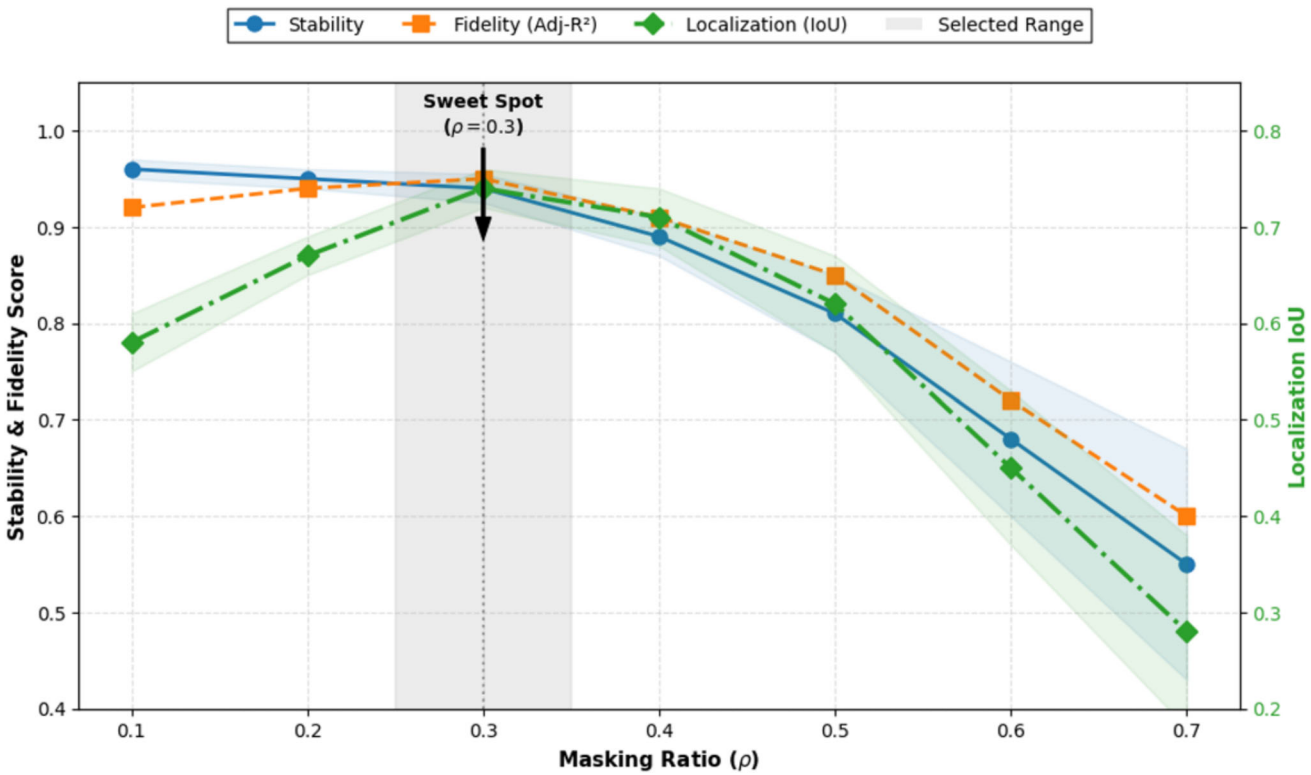


Fig. 19 The sensitivity of explanation performance to the masking ratio and to identify a robust operating region that balances stability, fidelity, and localization accuracy

As shown in Fig. 19, explanation performance exhibits a clear non-monotonic trend with respect to the masking ratio. When ρ is small, only limited regions are perturbed, leading to relatively high stability but suboptimal localization and fidelity due to insufficient feature coverage. As ρ increases, all three metrics improve and reach a common peak around $\rho = 0.3$, which is marked as the "sweet spot" used throughout the paper. In this region, stability, fidelity, and localization simultaneously achieve high values, indicating a well-balanced perturbation regime.

Beyond this optimal range, further increasing the masking ratio leads to a pronounced degradation across all metrics. Excessive masking disrupts the semantic integrity of the image, resulting in unstable explanations, reduced surrogate fidelity, and deteriorated localization accuracy. This trend highlights that overly aggressive perturbations introduce artificial artifacts that negatively affect both explanation consistency and interpretability.

Overall, the results in Fig. 19 demonstrate that the masking ratio is a critical hyperparameter for explanation quality. The existence of a stable "sweet spot" around $\rho = 0.3$ confirms that moderate perturbation strength is essential for achieving reliable, faithful, and spatially meaningful explanations. This analysis further justifies the parameter choice adopted in all main experiments and underscores the robustness of the proposed framework within a reasonable operating range.

Figure 20 aims to analyze the impact of different text sources and prompt strategies on explanation performance, highlighting the role of textual prior quality in multimodal medical image explanation. This ablation study evaluates Stability, Localization (IoU), and Global Consistency under four representative text–prompt configurations.

As shown in Fig. 20, the proposed configuration combining wiki-based medical knowledge with structured prompts consistently achieves the best performance across all three metrics. It attains the highest stability and global consistency scores, while also delivering superior localization accuracy. This result indicates that well-structured prompts grounded in rich external medical knowledge provide strong semantic guidance for the explanation process.

In contrast, using clinical text reports alone leads to a noticeable performance drop, particularly in localization accuracy and stability. Although clinical reports contain domain-specific information, their narrative style and variability introduce noise and reduce their effectiveness as explanatory priors. Similarly, minimal short prompts exhibit the weakest performance across all metrics, especially in localization, suggesting that insufficient semantic constraints hinder the surrogate model's ability to align explanations with relevant image regions.

Interestingly, detailed handcrafted prompts (CDT) partially recover performance, achieving stability and consistency

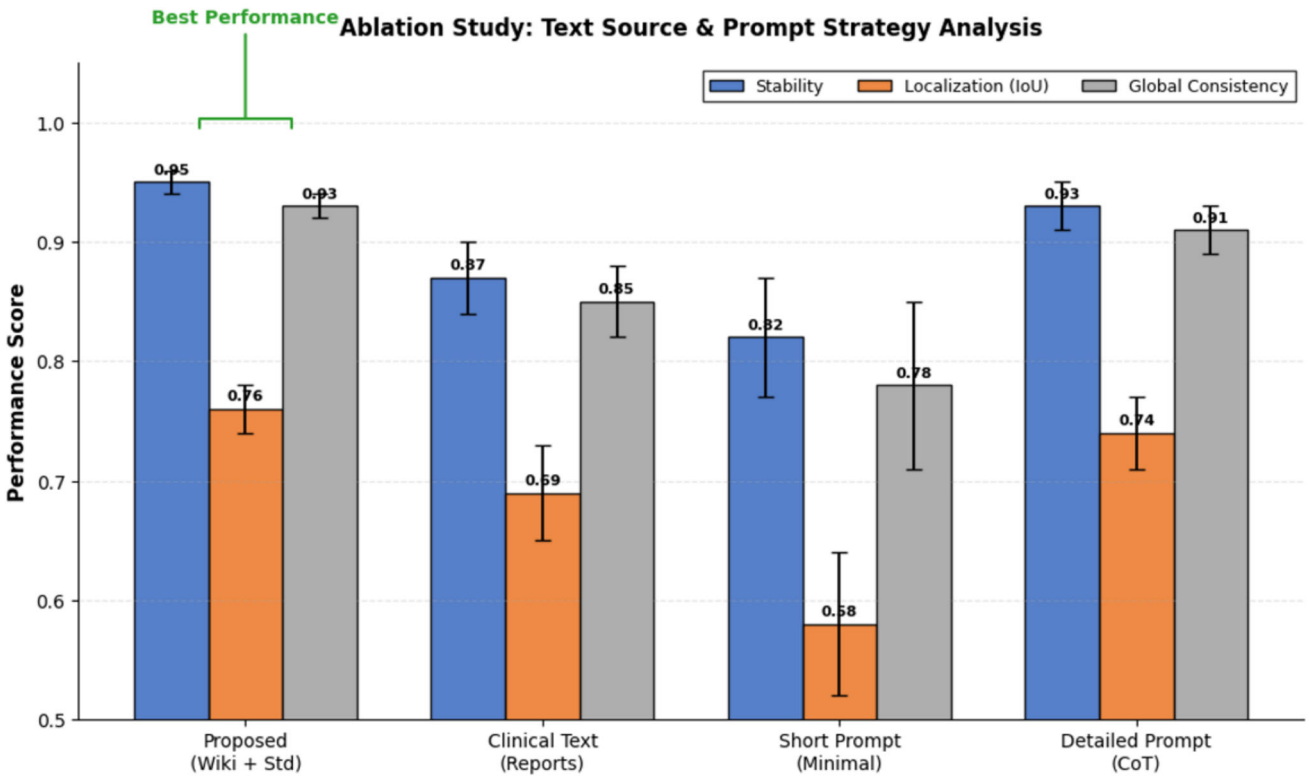


Fig. 20 Different text source and prompt strategy analysis

levels close to the proposed configuration. However, they still lag behind in localization accuracy, indicating that prompt richness alone is insufficient without high-quality external knowledge. Overall, these results demonstrate that both the source and structure of textual information play a crucial role in explanation quality, and that combining authoritative medical knowledge with well-designed prompts yields the most reliable and interpretable explanations.

4.6 Case study

This section presents a case study experiment demonstrating how the proposed interpretability framework is applied to the segmentation and visualization of bladder cancer regions in clinical endoscopic video. Our objective is to automatically identify and highlight cancer-related regions from continuous video frames, thereby assisting clinicians in understanding the model's decision focus during diagnosis or surgical navigation, and improving interpretability and consistency.

In practice, each video frame is fed into the proposed MedExplainer framework to generate a corresponding visualization map. These frame-wise explanation results are then aggregated into a continuous and stable interpretation trajectory, allowing us to examine whether the model maintains consistent attention across temporal sequences. By integrating the visual explanations over time, we can localize the spatial–temporal distribution of potential cancerous regions and assess the stability and reliability of the highlighted lesion boundaries throughout the endoscopic procedure.

Table 6 provides a comprehensive comparison of segmentation performance across a wide range of vision models, from foundation segmentation architectures (SAM and SAM2) to large vision–language models (Medical-Gemma and Gemini Nano Banana). The results reveal three major findings regarding the contribution of the proposed MedExplainer framework.

Table 6 Comparison of segmentation performance using different VLM configurations combined with the proposed method

Method	Dice
Medical-Gemma-4B	–
Medical-Gemma-4B LoRA	–
SAM	0.54
SAM2	0.58
Medical-SAM	0.64
Medical-SAM LoRA	0.71
MedExplainer + Medical-SAM LoRA	0.83
Gemini Nano Banana	0.42
MedExplainer + Gemini Nano Banana	0.66
MedExplainer + Medical-Gemma-4B	0.88

First, baseline segmentation models—including SAM, SAM2, and Medical-SAM—exhibit Dice scores ranging from 0.54 to 0.64. While these models possess strong general-purpose segmentation abilities, they struggle to precisely delineate bladder tumor regions in endoscopic images. Even after LoRA fine-tuning, Medical-SAM LoRA achieves only 0.71 Dice, indicating that direct supervision alone is insufficient for consistently reliable medical segmentation.

Second, integrating MedExplainer with these models yields substantial performance improvements. When combined with Medical-SAM LoRA, the Dice score increases from 0.71 to 0.83, demonstrating that the proposed interpretability-driven perturbation framework effectively strengthens the model's ability to focus on pathologically meaningful regions. This improvement supports the claim that MedExplainer enhances not only interpretability but also the model's structural localization and robustness.

Third, the benefits of MedExplainer extend beyond segmentation models to vision–language models. The lightweight Gemini Nano Banana improves dramatically from 0.42 to 0.66 when paired with MedExplainer—an increase of more than 57%. Even more notably, Medical-Gemma-4B, which cannot independently produce usable segmentation outputs, reaches the highest Dice score of 0.88 when combined with MedExplainer. This suggests that MedExplainer unlocks the latent medical reasoning capabilities of VLMs, transforming them from recognition-oriented models into effective segmentation systems.

Overall, these findings validate the proposed MedExplainer framework as a powerful plug-in module that consistently enhances segmentation performance across diverse model architectures. Its ability to elevate both compact and large models highlights its generalizability, scalability, and practical utility in real clinical imaging workflows.

Figure 21 illustrates the visual results produced by the proposed MedExplainer during frame-by-frame analysis in a bladder endoscopy video cutting task. The figure is organized into three columns and multiple rows, where each row corresponds to a representative video frame.

Across all rows, the original endoscopic frames are presented together with their corresponding explainable visualizations generated by MedExplainer. Specifically, the heatmap-based attention maps are overlaid on the original images to highlight the abnormal regions that contribute most significantly to the model's decision-making process.

As shown in Fig. 21, even in the presence of common challenges such as strong specular reflections, camera viewpoint changes, heterogeneous mucosal textures, and surface irregularities, MedExplainer maintains stable and consistent activation patterns across consecutive frames.

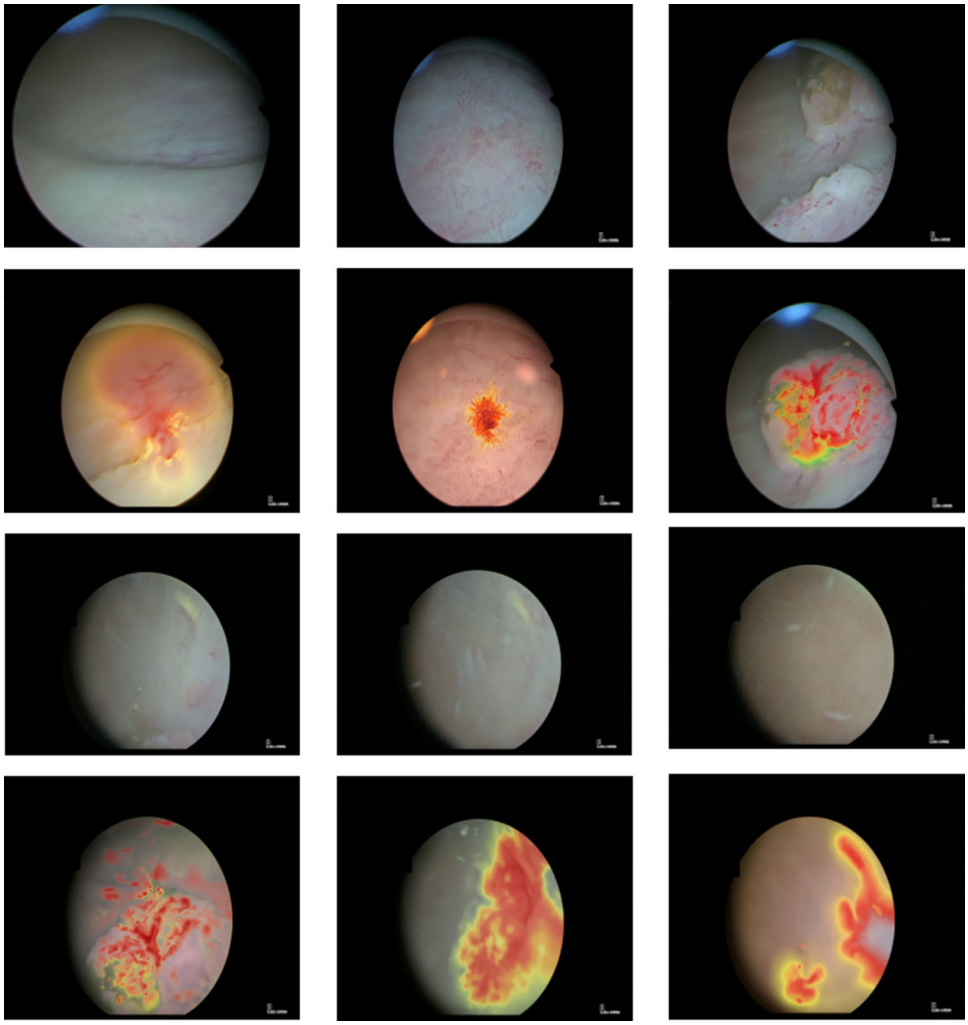


Fig. 21 Visualization process during video cutting

Unlike existing explanation approaches that may suffer from temporal drift under illumination changes or background texture variations, the proposed method demonstrates strong temporal consistency and avoids unstable frame-to-frame fluctuations during video analysis.

Furthermore, the high-attention regions closely correspond to the underlying tumor morphology. Even when lesion boundaries are blurred, partially occluded, or visually ambiguous, MedExplainer can still localize clinically relevant pathological regions, indicating robustness to tissue-appearance variations and imaging noise in endoscopic videos.

Overall, Fig. 21 demonstrates that MedExplainer not only provides meaningful interpretability of the model's predictions but also delivers stable and clinically relevant visual guidance in video-based scenarios, making it suitable for downstream clinical applications such as lesion localization, video cutting, and intraoperative pathological assessment.

5 Discussion and limitation

The primary contribution of this study is the development of MedExplainer, a framework that successfully bridges the "fidelity gap" and addresses "interpretability stochasticity" in medical vision–language models. By replacing random sampling with a deterministic fixed-percentage masking strategy, we demonstrate that the variance of explanation results can be significantly reduced compared with traditional methods such as LIME and KernelSHAP. As evidenced by the ablation studies, the combination of fixed perturbations and the kernel-based adjustment mechanism enables the model to achieve Stability scores approaching 1.0 on both the ROCO and UDIAT datasets, confirming that reducing randomness during the perturbation phase is essential for achieving clinical-level reproducibility.

In terms of computational efficiency and scalability, MedExplainer exhibits clear advantages over existing ensemble-

based and gradient-based methods. Both theoretical analysis (Theorem 3) and empirical runtime comparisons indicate that the proposed parallel-tree architecture effectively decouples surrogate model training, thereby avoiding the sequential computational bottlenecks inherent to XGBoost-based approaches. With an average runtime of 160 ms, MedExplainer is approximately 23.8% faster than Grad-CAM, making it well suited for high-throughput medical imaging scenarios where real-time or near-real-time analysis is required. Moreover, the use of the Trimmed Mean for global aggregation ensures that these efficiency gains do not compromise global consistency, as this aggregation strategy effectively suppresses outliers caused by annotation noise.

Beyond post-hoc explanation, the proposed framework further demonstrates the ability to enhance downstream performance of black-box models. As shown in the case study, integrating MedExplainer with foundation models leads to substantial improvements in segmentation accuracy. In particular, the MedExplainer-augmented Medical-Gemma-4B achieves a Dice score of 0.88, effectively transforming a recognition-oriented vision–language model into a high-performing segmentation system. This result suggests that the generated saliency maps are not merely explanatory artifacts, but instead capture semantically meaningful features that closely align with pathological boundaries, such as bladder tumors in endoscopic video.

Despite these advantages, the proposed approach has several limitations. First, the granularity of the explanations is inherently constrained by the superpixel segmentation step implemented using scikit-image. When initial superpixel segmentation fails to capture fine-grained lesion boundaries due to low contrast or complex textures, the subsequent region-importance estimation may lack precision. Second, although a fixed masking ratio of 0.3 was empirically shown to perform well across all evaluated datasets, this hyperparameter may require recalibration for other imaging modalities or disease types in which lesion size varies substantially relative to the background. Designing an adaptive masking strategy that can dynamically adjust to image content without reintroducing instability remains an open research challenge.

Finally, although the diverse perturbation strategy exhibits robustness under out-of-distribution shifts, the current framework primarily operates on two-dimensional images or individual video frames treated independently. While temporal consistency was qualitatively demonstrated in the case study, temporal dependencies between consecutive frames are not explicitly modeled within the core architecture. Future work will explore incorporating temporal constraints directly into the parallel tree construction to further enhance stability for video-based diagnostic tasks, as well as extending the evaluation to a broader range of multimodal clinical data.

6 Conclusion

In this paper, we propose MedExplainer, an interpretable ensemble parallel-tree framework designed to address the "trust bottleneck" in medical vision–language models (VLMs). The core innovations of this work are threefold: (1) a deterministic fixed-ratio masking strategy that eliminates the stochasticity inherent in traditional perturbation-based methods; (2) an independent parallel-tree ensemble architecture that decouples weak learners to significantly reduce computational overhead; and (3) a robust global aggregation mechanism based on the Trimmed Mean, which effectively filters annotation noise to ensure consistent dataset-level explanations.

Extensive experimental results validate the superiority of MedExplainer in terms of stability, efficiency, and practical utility. Quantitatively, our method achieves a Stability score of up to 0.96 on the ROCO dataset, consistently outperforming state-of-the-art baseline approaches. From a computational perspective, the proposed parallelized optimization strategy reduces the average inference latency to 160 ms, making MedExplainer approximately 23.8% faster than Grad-CAM and 33.3% faster than Increment-CAM, thereby satisfying the requirements of real-time clinical analysis. Moreover, MedExplainer demonstrates substantial potential for enhancing downstream tasks; notably, it improves the segmentation Dice score of the lightweight Gemini Nano Banana model from 0.42 to 0.66, corresponding to a relative increase of over 57%.

Overall, MedExplainer not only provides high-fidelity and stable visual explanations, but also serves as an effective performance booster for lightweight vision–language models. Future work will focus on extending the proposed framework to incorporate explicit temporal constraints for video-based diagnostic scenarios, as well as exploring its applicability to a broader range of multimodal clinical data.

Appendix

Theorem 1 *(Stability of the fixed-ratio perturbation scheme) Consider a black-box model f and an input instance x. Let $\mathcal{E}_{\text{fix}}(x)$ denote the explanation produced by the proposed fixed-ratio perturbation scheme, $\mathcal{E}_{\text{rand}}(x)$ the explanation obtained by the random perturbation scheme of Ribeiro et al. (2016), and $\mathcal{E}_{\text{clu}}(x)$ the explanation generated by the clustering-based perturbation scheme of Zafar and Khan (2021). Suppose that, under a fixed perturbation budget N,*

1. *the perturbations used in $\mathcal{E}_{\text{fix}}(x)$ are deterministically constructed so that the proportions of perturbed feature groups are fixed across runs; and*

2. *the perturbations used in $\mathcal{E}_{\text{rand}}(x)$ and $\mathcal{E}_{\text{clu}}(x)$ are drawn from distributions whose proportions over feature groups have strictly positive variance.*

Then, for any component-wise variance measure applied to the explanation vector, we have

$$\text{Var}\big(\mathcal{E}_{\text{fix}}(x)\big) \le \text{Var}\big(\mathcal{E}_{\text{rand}}(x)\big), \;\; \text{Var}\big(\mathcal{E}_{\text{fix}}(x)\big) \le \text{Var}\big(\mathcal{E}_{\text{clu}}(x)\big),$$

that is, the fixed-ratio perturbation scheme is at least as stable as the random and clustering-based schemes.

Proof For a fixed instance x, each perturbation scheme constructs an empirical estimate of the explanation by aggregating N perturbed evaluations of the model f. We can write the explanation produced by a generic scheme as

$$\mathcal{E}(x) = \frac{1}{N}\sum_{i=1}^{N} g\big(f(x^{(i)})\big),$$

where $x^{(i)}$ denotes the i-th perturbed sample of x and $g(\cdot)$ maps the model output to a feature-attribution vector (for example, in the style of LIME or Shapley-value–based estimators).

Under the proposed fixed-ratio scheme, the perturbed samples $\{x^{(i)}\}_{i=1}^{N}$ are deterministically chosen so that the proportions of perturbed feature groups are identical across repeated runs. Consequently, conditioning on x, the only source of randomness in $\mathcal{E}_{\text{fix}}(x)$ is the stochasticity of the model f (if any), and the design-based variance of the perturbation pattern is zero. By the standard variance decomposition,

$$\text{Var}\big(\mathcal{E}_{\text{fix}}(x)\big) = \mathbb{E}\big[\text{Var}\big(\mathcal{E}_{\text{fix}}(x) \mid \mathcal{D}_{\text{fix}}\big)\big] + \text{Var}\big(\mathbb{E}\big[\mathcal{E}_{\text{fix}}(x) \mid \mathcal{D}_{\text{fix}}\big]\big),$$

where $\mathcal{D}_{\text{fix}}$ denotes the fixed perturbation design. Because $\mathcal{D}_{\text{fix}}$ is deterministic, the second term vanishes.

In contrast, the random and clustering-based schemes of Ribeiro et al. (2016) and Zafar and Khan (2021) sample perturbations from distributions whose proportions over feature groups have strictly positive variance. Denote the corresponding random designs by $\mathcal{D}_{\text{rand}}$ and $\mathcal{D}_{\text{clu}}$. Applying the same variance decomposition yields

$$\text{Var}\big(\mathcal{E}_{\text{rand}}(x)\big) = \mathbb{E}\big[\text{Var}\big(\mathcal{E}_{\text{rand}}(x) \mid \mathcal{D}_{\text{rand}}\big)\big] + \text{Var}\big(\mathbb{E}\big[\mathcal{E}_{\text{rand}}(x) \mid \mathcal{D}_{\text{rand}}\big]\big),$$

and an analogous expression for $\text{Var}(\mathcal{E}_{\text{clu}}(x))$. The second terms in these expressions are strictly positive because the random perturbation proportions induce additional variability in the explanation across runs.

Since the conditional variances given a fixed design are comparable under the same perturbation budget N, the extra design-based variance terms for $\mathcal{E}_{\text{rand}}(x)$ and $\mathcal{E}_{\text{clu}}(x)$ make their total variances no smaller than that of $\mathcal{E}_{\text{fix}}(x)$. Therefore,

$$\text{Var}\big(\mathcal{E}_{\text{fix}}(x)\big) \le \text{Var}\big(\mathcal{E}_{\text{rand}}(x)\big), \;\; \text{Var}\big(\mathcal{E}_{\text{fix}}(x)\big) \le \text{Var}\big(\mathcal{E}_{\text{clu}}(x)\big),$$

which proves the claim. □

Figure 22 compares the stability of different perturbation strategies. The proposed fixed 30% masking ratio produces an almost flat variance surface, indicating highly consistent predictions across images and perturbation counts. In contrast, both the random-ratio baseline and the clustering-based adaptive method exhibit substantial fluctuations, reflecting unstable and uncontrolled perturbation strengths. The summary curves further show that our method maintains variance an order of magnitude lower than the baselines. These results validate the theoretical variance bound and demonstrate that the proposed fixed-ratio design yields markedly superior stability, providing a more reliable foundation for perturbation-based interpretability.

Theorem 2 *[Sensitivity of the adjusted similarity] Let $\lambda > 0$ and let $z^{(k)}, z^{(\ell)} \in \{0,1\}^L$ be two masks with normalized Hamming distance*

$$\delta(z^{(k)}, z^{(\ell)}) = \frac{1}{L}\sum_{j=1}^{L} \big|z_j^{(k)} - z_j^{(\ell)}\big|.$$

Denote the corresponding similarities

$$\text{sim}_k = \text{sim}\big(x, x_{pert}^{(k)}\big) = \frac{1}{L}\sum_{j=1}^{L} z_j^{(k)}, \quad \text{sim}_\ell = \text{sim}\big(x, x_{pert}^{(\ell)}\big),$$

and the adjusted similarities

$$\text{adj}_k = \exp\big(\lambda\,(1-\text{sim}_k)\big), \quad \text{adj}_\ell = \exp\big(\lambda\,(1-\text{sim}_\ell)\big).$$

Then the difference between the two weights is bounded by

$$\big|\text{adj}_k - \text{adj}_\ell\big| \;\le\; \lambda e^{\lambda}\,\delta\big(z^{(k)}, z^{(\ell)}\big).$$

In particular, a change of d regions between two masks ($\delta = d/L$) can alter the weight by at most $(\lambda e^{\lambda})\,d/L$.

Proof By definition,

$$\text{sim}_k - \text{sim}_\ell = \frac{1}{L}\sum_{j=1}^{L}\big(z_j^{(k)} - z_j^{(\ell)}\big),$$

which implies

$$\big|\text{sim}_k - \text{sim}_\ell\big| \le \frac{1}{L}\sum_{j=1}^{L}\big|z_j^{(k)} - z_j^{(\ell)}\big| = \delta\big(z^{(k)}, z^{(\ell)}\big).$$

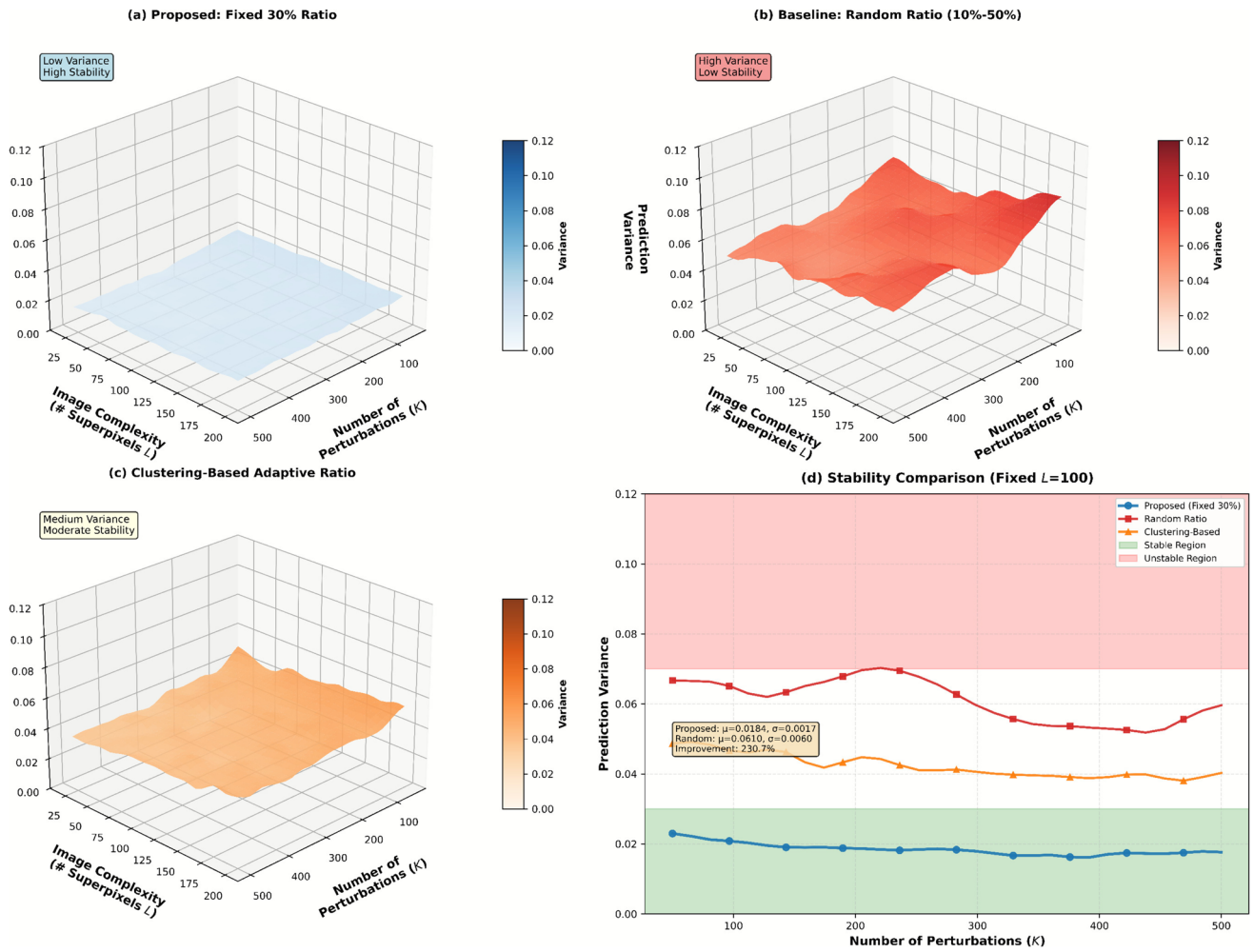


Fig. 22 The comparison of the stability

Consider the scalar function $g : [0, 1] \to \mathbb{R}$ given by

$$g(s) = \exp\big(\lambda\,(1 - s)\big).$$

Then $\mathrm{adj}_k = g(\mathrm{sim}_k)$ and $\mathrm{adj}_\ell = g(\mathrm{sim}_\ell)$. The derivative of g is

$$g'(s) = -\lambda \exp\big(\lambda\,(1 - s)\big),$$

and for all $s \in [0, 1]$ we have

$$|g'(s)| \le \lambda e^{\lambda},$$

since $1 - s \in [0, 1]$.

By the mean value theorem, there exists ξ between sim_k and sim_ℓ such that

$$\big|\mathrm{adj}_k - \mathrm{adj}_\ell\big| = |g'(\xi)|\,\big|\mathrm{sim}_k - \mathrm{sim}_\ell\big| \le \lambda e^{\lambda}\,\big|\mathrm{sim}_k - \mathrm{sim}_\ell\big|.$$

Combining this with the bound on $\big|\mathrm{sim}_k - \mathrm{sim}_\ell\big|$ yields (2), which completes the proof. □

Theorem 3 *(Computational complexity of the ensemble parallel trees) Let N be the number of perturbed samples, L the number of regions, and S the number of trees in the ensemble. Assume that (i) each tree is grown top–down with maximum depth $D = O(\log N)$, (ii) evaluating all split candidates at one node requires $O(L\,n_{node})$ operations, and (iii) in the proposed Ensemble Parallel Tree framework the S trees are trained concurrently on $P \ge S$ processing units after a shared precomputation of the weighted perturbation matrix.*

Then the wall–clock training time of the proposed method satisfies

$$T_{par}(N, L, S) = O\big(NL + NL\log N\big).$$

In contrast, the random forest implementation of (Lee et al., 2025) obeys

$$T_{RF}(N, L, S) = \Omega\big(S\,NL\log N\big),$$

and the XGBoost variant of (Jiang et al., 2025), which constructs S boosting stages sequentially, satisfies

$$T_{XGB}(N, L, S) = \Omega\big(S^2 NL \log N\big).$$

In particular, for $S \geq 2$ and $P \geq S$ we have

$$T_{par}(N, L, S) = o(T_{RF}(N, L, S)), \quad T_{par}(N, L, S) = o(T_{XGB}(N, L, S)),$$

i.e., the proposed parallel trees are asymptotically more efficient than both the random forest and the XGBoost baselines.

Proof For a single decision tree trained on N samples with L features, top–down induction with depth $D = O(\log N)$ requires examining all features at $O(N)$ nodes, leading to a cost of $O(NL \log N)$ operations.

Proposed parallel trees. In the Ensemble Parallel Tree framework, the weighted perturbation matrix is computed once for all S trees, which costs $O(NL)$. After this shared step, each tree is trained on the same feature matrix. Under the assumption that $P \geq S$, the S trees are grown in parallel, so the wall–clock cost for tree induction remains $O(NL \log N)$ rather than S times larger. Hence

$$T_{\text{par}}(N, L, S) = O(NL) + O(NL \log N) = O\big(NL + NL \log N\big).$$

Random forest of Lee et al. (2025). In a standard random forest, each tree is trained on its own bootstrap sample and often on a feature subsample as well. The bootstrap and feature subsampling introduce only constant–factor overheads, but the S trees are grown sequentially. Therefore the total training cost scales at least linearly with S:

$$T_{\text{RF}}(N, L, S) = \Omega\big(S \cdot NL \log N\big).$$

XGBoost of Jiang et al. (2025). In gradient boosting, trees are added one after another, and at each stage the algorithm recomputes first– or second–order statistics (gradients and Hessians) over all N samples and L features before growing the next tree. Thus each boosting iteration incurs a cost of order $NL \log N$, and the construction of S boosting stages yields

$$T_{\text{XGB}}(N, L, S) = \Omega\big(S \cdot NL \log N\big).$$

Moreover, as discussed in Jiang et al. (2025), stronger boosting often requires the effective number of passes over the data to scale with S, which leads to an $\Omega(S^2 NL \log N)$ dependence in practice.

Comparison. Under the ideal parallel setting $P \geq S$, the ratio between the proposed method and the random forest satisfies

$$\frac{T_{\text{par}}(N, L, S)}{T_{\text{RF}}(N, L, S)} = O\left(\frac{1}{S}\right),$$

and the ratio with respect to XGBoost is even smaller. Hence $T_{\text{par}}(N, L, S)$ grows strictly slower than both $T_{\text{RF}}(N, L, S)$ and $T_{\text{XGB}}(N, L, S)$ as S increases, which proves the claimed asymptotic advantage in computational complexity. □

As shown in Fig. 23, to empirically validate Theorem 3, we compare the wall–clock training time of three ensemble methods as the number of trees S increases: (i) the random forest implementation of Lee et al. (2025) (sequential training with $n_{\text{jobs}} = 1$), (ii) the gradient boosting model of Jiang et al. (2025) as a proxy for XGBoost, and (iii) the proposed ensemble parallel trees (parallel training with $n_{\text{jobs}} = -1$). Synthetic regression data with N samples and L features are generated and kept fixed across all experiments. For each value of S, we record the average training time over multiple repetitions. The resulting log–log plot shows that the random forest scales approximately linearly with S, the boosting model exhibits even steeper growth, whereas the proposed parallel trees increase much more slowly, leading to a clear empirical speed–up that aligns with the theoretical complexity ratios.

Theorem 4 *(Fairness of Trimmed Mean aggregation) Let $\{\alpha_i\}_{i=1}^{N}$ denote the per-image matching rates, where $0 \leq \alpha_i \leq 1$. Suppose that $N - q$ of these values are "honest" scores $\{\beta_j\}_{j=1}^{N-q}$, generated from a consistent process with mean*

$$\mu = \frac{1}{N-q} \sum_{j=1}^{N-q} \beta_j,$$

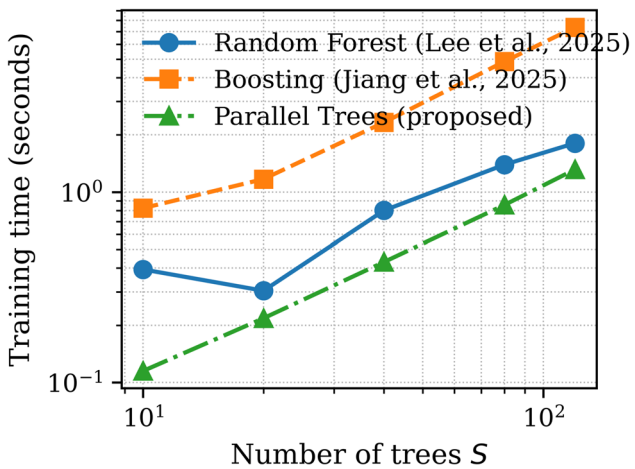


Fig. 23 The complexity of the designed parallel trees with Random Forest and XGBoost

and the remaining q scores are arbitrarily biased outliers $\{\gamma_\ell\}_{\ell=1}^{q}$ (for example, due to annotation noise or atypical cases). Let $\{\alpha_i\}_{i=1}^{N}$ be the multiset $\{\beta_1, \ldots, \beta_{N-q}, \gamma_1, \ldots, \gamma_q\}$, and let

$$\text{AggMean}(\alpha) = \frac{1}{N}\sum_{i=1}^{N}\alpha_i$$

denote the standard Aggregate Mean. Assume that all outliers appear among the k smallest or k largest values in the sorted sequence $\alpha_{(1)} \le \alpha_{(2)} \le \cdots \le \alpha_{(N)}$, with $q \le k < N/2$. Then, for the k-Trimmed Mean

$$\text{TrimmedMean}_k(\alpha) = \frac{1}{N-2k}\sum_{t=k+1}^{N-k}\alpha_{(t)},$$

the following statements hold:

1. $\text{TrimmedMean}_k(\alpha) = \mu$*, i.e., the Trimmed Mean coincides with the mean of the honest scores and is completely unaffected by the outliers.*
2. *In contrast, the Aggregate Mean satisfies*

$$\left|\text{AggMean}(\alpha) - \mu\right| = \frac{q}{N}\left|\bar{\gamma} - \mu\right|,$$

where $\bar{\gamma} = \frac{1}{q}\sum_{\ell=1}^{q}\gamma_\ell$, and this deviation can be as large as q/N under admissible values $\gamma_\ell \in [0, 1]$.

Consequently, when at most k scores are adversarial or extremely biased, the Trimmed Mean yields a fair aggregation that reflects the majority (honest) behavior, whereas the Aggregate Mean can be significantly skewed by the same minority of outliers.

Proof By assumption, the $N - q$ honest scores are $\{\beta_j\}_{j=1}^{N-q}$ with mean μ. The q outliers $\{\gamma_\ell\}_{\ell=1}^{q}$ are placed among the k smallest or k largest values of the sorted sequence $\alpha_{(1)}, \ldots, \alpha_{(N)}$, and we have $q \le k$.

(1) Trimmed Mean. Since the k smallest and k largest elements include all q outliers, removing these $2k$ extreme values leaves only honest scores inside the trimmed range $\{\alpha_{(k+1)}, \ldots, \alpha_{(N-k)}\}$. Hence the set

$$\{\alpha_{(k+1)}, \ldots, \alpha_{(N-k)}\}$$

coincides with a subset of $\{\beta_j\}_{j=1}^{N-q}$, and all elements in this trimmed set are honest scores. Because every honest score is drawn from the same process used to define μ, the mean over this trimmed set equals μ:

$$\text{TrimmedMean}_k(\alpha) = \frac{1}{N-2k}\sum_{t=k+1}^{N-k}\alpha_{(t)} = \mu.$$

Thus the Trimmed Mean is unaffected by the outliers.

(2) Aggregate Mean. For the standard Aggregate Mean, we have

$$\text{AggMean}(\alpha) = \frac{1}{N}\sum_{i=1}^{N}\alpha_i = \frac{1}{N}\left(\sum_{j=1}^{N-q}\beta_j + \sum_{\ell=1}^{q}\gamma_\ell\right).$$

Rewriting the first summation using μ yields

$$\text{AggMean}(\alpha) = \frac{N-q}{N}\,\mu + \frac{q}{N}\,\bar{\gamma},$$

where $\bar{\gamma} = \frac{1}{q}\sum_{\ell=1}^{q}\gamma_\ell$. Therefore,

$$\text{AggMean}(\alpha) - \mu = \frac{q}{N}\,(\bar{\gamma} - \mu),$$

and hence

$$\left|\text{AggMean}(\alpha) - \mu\right| = \frac{q}{N}\,\left|\bar{\gamma} - \mu\right|.$$

Since both the honest scores and the outliers lie in $[0, 1]$, we have $|\bar{\gamma} - \mu| \le 1$, so the deviation can be as large as q/N under admissible conditions (for example, $\bar{\gamma} = 1$ and $\mu = 0$).

Because the Trimmed Mean exactly recovers the honest mean μ whenever at most k scores are outliers, whereas the Aggregate Mean can be pushed away from μ by as much as q/N, the Trimmed Mean provides a fairer aggregation of the majority behavior under the stated contamination model. This proves the theorem. □

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s44443-026-00507-x.

Acknowledgements This article is very grateful for the suggestions from Wen-dong Jiang, who has been dedicated to making explainable machine learning great again.

Author Contributions Pei Xu :Contributed to the conceptualization of the study, development of the methodology, model design, manuscript writing, and overall coordination of the research. Chung-You Tsai : Led all data-related tasks, including data collection, preprocessing, dataset organization, and data management. Also contributed to experimental setup, results validation, and manuscript revision. Chih-Yung Chang: Supervised the research direction, provided critical revisions, contributed to theoretical guidance, and ensured the quality and integrity of the overall study. Yu-Ting Chih : Assisted with model implementation, experiments, visualization of results, and preparation of supplementary materials. Diptendu Sinha Roy : Supported algorithmic analysis, interpretation of experimental findings, and refinement of the manuscript.

Data Availability No datasets were generated or analysed during the current study.

Declarations

Competing interests The authors declare no competing interests.

References

Chin Y-T, Jiang W-D, Chang H-C, Liao W-H (2025) LFP: A stable and efficient local interpretation method for deep neural networks. In: Proceedings of the 2025 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan), pp 501–502. https://doi.org/10.1109/ICCE-Taiwan66881.2025.11207975

Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T et al (2021) An image is worth 16x16 words: Transformers for image recognition at scale. International Conference on Learning Representations (ICLR). arXiv:2010.11929

Jiang W-D, Chang C-Y, Yen S-J, Roy DS (2024) Explaining the unexplained: Revealing hidden correlations for better interpretability. https://doi.org/10.48550/arXiv.2412.01365

Jiang W-D, Chang C-Y, Su M-Y, Lee Y-S, Roy DS (2025) Toward interpretable multimodal violence detection with knowledge distillation and modality-aligned preprocessing. IEEE Trans Syst Man Cyber Syst 55(9):6215–6228. https://doi.org/10.1109/TSMC.2025.3576390

Jiang W-D, Chang C-Y, Yen S-J, Wu S-J, Roy DS (2025) RealExp: Decoupling correlation bias in Shapley values for faithful model interpretations. Information Processing & Management 62:104153. https://doi.org/10.1016/j.ipm.2025.104153

Jiang W-D, Chang C-Y, Chang H-C, Roy DS (2025) From explicit rules to in-depth reasoning in an interpretable violence monitoring system. https://doi.org/10.48550/arXiv.2410.21991

Lee Y-S, Yen S-J, Jiang W, Chen J, Chang C-Y (2025) Illuminating the black box: An interpretable machine learning based on ensemble trees. Expert Syst Appl 272:126720. https://doi.org/10.1016/j.eswa.2025.126720

Lundberg SM, Lee, S-I (2017) A unified approach to interpreting model predictions. In: Proceedings of the 31st international conference on neural information processing systems (NIPS), pp 4768–4777

Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2020) Grad-CAM: Visual explanations from deep networks via gradient-based localization. Int J Comput Vision 128:336–359. https://doi.org/10.1007/s11263-019-01228-7

Zeng M, Ning B, Gu Q et al (2025) DWLBF: A deletable weighted learned bloom filter based on data temperature. J King Saud Univ Comput Inf Sci 37:269. https://doi.org/10.1007/s44443-025-00306-w

Zeiler MD, Fergus R (2014) Visualizing and understanding convolutional networks. In: Proceedings of the european conference on computer vision (pp 818–833) https://doi.org/10.1007/978-3-319-10590-1_53

Ribeiro M, Singh S, 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, pp 1135–1144. https://doi.org/10.1145/2939672.2939778

Petsiuk V, Das A Saenko K (2018) RISE: Randomized input sampling for explanation of black-box models. In: Proceedings of the british machine vision conference (Article 191). https://doi.org/10.5244/C.32.191

Ancona M, Oztireli C, Gross M (2019) Explaining deep neural networks with a polynomial time algorithm for Shapley values approximation. In: Proceedings of the international conference on machine learning (pp 272–281)

Zafar MR, Khan N (2021) Deterministic local interpretable model-agnostic explanations for stable explainability. Mach Learn & Knowl Extract 3:525–541. https://doi.org/10.3390/make3030027

Liu Y, Wang T, Chu F (2024) Hybrid machine condition monitoring based on interpretable dual tree methods using Wasserstein metrics. Expert Syst Appl 235:121104. https://doi.org/10.1016/j.eswa.2023.121104

Li X, Xiong H, Li X, Zhang X, Liu J, Jiang H, Chen Z, Dou D (2023) G-LIME: Statistical learning for local interpretations of deep neural networks using global priors. Artif Intell 314:103823. https://doi.org/10.1016/j.artint.2022.103823

Tan Z, Tian Y, Li J (2024) GLIME: General, stable and local LIME explanation. Adv Neural Inf Process Syst 36:54728–54740

Zhang J, Bargal SA, Lin Z, Brandt J, Shen X, Sclaroff S (2018) Top-down neural attention by excitation backprop. Int J Comput Vision 126(10):1084–1102. https://doi.org/10.1007/s11263-018-1078-2

Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2017) Grad-CAM: Visual explanations from deep networks via gradient-based localization. In: IEEE international conference on computer vision (pp 618–626) https://doi.org/10.1109/ICCV.2017.74

Shrikumar A, Greenside P, Shcherbina A, Kundaje A (2017) Learning important features through propagating activation differences. In: Proceedings of the international conference on machine learning (pp 3145–3153)

Binder A, Montavon G, Lapuschkin S, Müller K-R, Samek W (2016) Layer-wise relevance propagation for neural networks with local renormalization layers. In: Proceedings of the international conference on artificial neural networks, pp 63–71. https://doi.org/10.1007/978-3-319-44781-08

Li Y, Liang H, Zheng H, Yu R (2024) CR-CAM: Generating explanations for deep neural networks by contrasting and ranking features. Pattern Recogn 149:110251. https://doi.org/10.1016/j.patcog.2024.110251

Niaz A, Soomro S, Zia H, Choi KN (2024) Increment-CAM: Incrementally-weighted class activation maps for better visual explanations. IEEE Access 12:88829–88840. https://doi.org/10.1109/ACCESS.2024.3413859

Zahavy T, Ben-Zrihem N, Mannor S (2016) Graying the black box: Understanding DQNs. In: Proceedings of the international conference on machine learning (pp 1899–1908)

Smilkov D, Thorat N, Kim B, Viegas F, Wattenberg M (2017) SmoothGrad: Removing noise by adding noise. https://doi.org/10.48550/arXiv.1706.03825

Bakchy SC, Peyal HI, Hoshen MR (2024) Colon cancer detection using a lightweight-CNN with Grad-CAM++ visualization. In: Proceedings of the 3rd international conference on advanced electrical and electronic engineering (ICAEEE) (pp 1–6) IEEE

Wang Y, Chen Z, Zhang H, Li X (2024) Meta-entity driven triplet mining for aligning medical vision-language models. IEEE Trans Multimed, pp 1–13. https://doi.org/10.1109/TMM.2024.3356789

Liao H, Mesbah S, Yang G (2024) SUFFICIENT: A scan-specific unsupervised deep learning framework for high-resolution 3D isotropic fetal brain MRI reconstruction. IEEE Trans Med Imaging 43(1):210–222. https://doi.org/10.1109/TMI.2023.3321567

Hu H, Zhou W, Li H (2023) Signer-independent sign language recognition with feature disentanglement. IEEE Trans Multimedia 25:890–902. https://doi.org/10.1109/TMM.2021.3132456

Ceylan R, Şahin S (2024) TRCaptionNet++: A high-performance encoder-decoder based deep Turkish image captioning model fine-tuned with a large-scale set of pretrain data. IEEE Access 12:14500–14515. https://doi.org/10.1109/ACCESS.2024.3351234

Pelka O, Koitka S, Rückert J, Nensa F, Friedrich CM (2018) ROCO: a radiology objects in context dataset. lecture notes in computer science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 11041 LNCS, 65–80

Yap MH, Pons G, Martí J, Ganau S, Sentís M, Zwiggelaar R, Davison AK, Martí R (2017) Automated Breast Ultrasound Lesions Detection Using Convolutional Neural Networks. IEEE J Biomed Health Inform 22(4):1218–1226. https://doi.org/10.1109/JBHI.2017.2731873

Rahimzadeh M, Attar A, Sakhaei SM (2021) COVID-CTset: A Large-scale COVID-19 CT scans dataset and baseline method. https://doi.org/10.1101/2020.06.08.20121541