Explainability AI for Deep CNNs: A Comparative Study of Grad-CAM, LIME, and SHAP on CIFAR-10

Adrien Grevet
ESILV
adrien.grevet@edu.devinci.fr

January 20, 2026

Course: CV and Deep Learning
Head of Course: Ruiwen HE
Project Supervisor: Pierre LEFEBVRE

Abstract

This report presents a comparative study of post-hoc explainability methods for deep convolutional neural networks, conducted in the context of the "CV and Deep Learning" course. We train a VGG-16 model adapted to CIFAR-10 and compare three complementary explanation approaches: Grad-CAM (gradient-based), LIME (perturbation-based local surrogate), and SHAP (gradient-based attribution).

To evaluate explanations beyond visual appeal, we measure (i) faithfulness with a deletion protocol based on masking the most important pixels in 10 steps (AUC; lower is better) and (ii) efficiency as wall-clock runtime per explanation. The classifier reaches 92.79% test accuracy. Grad-CAM is the fastest method (0.083s) but yields higher deletion AUC (0.705), LIME is slower (0.384s) with similar deletion AUC (0.692), and SHAP achieves the best faithfulness (0.267) at the highest cost (1.386s). We complement these results with qualitative galleries on correct and incorrect predictions.

Contents

1 Introduction

Deep convolutional neural networks (CNNs) have become the dominant approach for visual recognition, achieving high accuracy across a wide range of image classification tasks. However, these gains often come at the cost of interpretability: modern CNNs are frequently treated as "black boxes", making it difficult to understand why a model predicts a given class or whether it relies on robust, semantically meaningful evidence.

This lack of transparency is not merely an academic concern. In safety- and trust-critical settings (e.g., medical imaging, autonomous driving, and industrial inspection), stakeholders must be able to audit model behavior, detect spurious correlations, and justify decisions. Explainable Artificial Intelligence (XAI) addresses these needs by providing post-hoc explanations that highlight which parts of the input contribute to a prediction.

In this project, we focus on three widely used post-hoc explainability methods that represent complementary design philosophies. Grad-CAM produces class-discriminative heatmaps using gradients within the network, enabling fast and intuitive visual inspection [1]. LIME approximates a model locally by perturbing inputs and fitting an interpretable surrogate model, offering model-agnostic explanations over image regions [2]. SHAP provides theoretically grounded feature attributions derived from Shapley values; we use a gradient-based SHAP explainer to obtain pixel-level (or region-level) importance maps [3].

While qualitative heatmaps can be compelling, they can also be misleading if not evaluated for faithfulness. Explanations that look plausible to humans may not reflect the evidence the model actually uses. To address this, we adopt a comparative evaluation framework that combines qualitative inspection with quantitative metrics.

2 Related Work

Deep neural networks have achieved state-of-the-art performance in computer vision, but their lack of transparency has raised concerns for deployment in high-stakes domains. Explainable AI (XAI) addresses this issue by producing post-hoc explanations that aim to (i) highlight which parts of an input drive a prediction and (ii) provide evidence that the prediction is based on semantically meaningful features rather than spurious correlations.

2.1 Gradient-based visual explanations

Early work on visual explanations for convolutional networks relied on gradient saliency, i.e., the sensitivity of a class score to changes in the input. While saliency maps are straightforward to compute, they are often noisy and can be difficult to interpret. A key refinement is to move from pixel-level gradients to explanations that operate on convolutional feature maps, which carry higher-level spatial semantics.

Grad-CAM is a widely used gradient-based method that produces class-discriminative localization maps by weighting the channels of a selected convolutional layer using gradients of the target class with respect to those feature maps, followed by a ReLU to focus on features that positively

influence the class score [1]. Compared to raw saliency, Grad-CAM typically yields smoother and more human-interpretable heatmaps, and it can be applied to a broad range of CNN architectures without retraining.

2.2 Model-agnostic, perturbation-based explanations

A different family of methods explains predictions by perturbing the input and observing changes in the model output. These approaches are attractive because they treat the classifier as a black box and can be used across model classes, including deep CNNs.

LIME (Local Interpretable Model-agnostic Explanations) fits a simple surrogate model (commonly a sparse linear regressor) in the neighborhood of a specific instance, using perturbed samples to approximate the local decision boundary [2]. For images, the perturbations are typically defined over super-pixels rather than individual pixels to keep the explanation components interpretable. LIME explanations can be informative but are sensitive to choices such as the super-pixel segmentation, the sampling strategy, and regularization strength, which may affect stability across runs.

SHAP provides an alternative perspective by grounding feature attributions in cooperative game theory: Shapley values measure the marginal contribution of each feature across all possible coalitions [3]. In practice, exact Shapley values are intractable for high-dimensional images, so approximations such as KernelSHAP are used. KernelSHAP estimates Shapley values by sampling feature subsets and using a weighted regression that satisfies desirable axioms (e.g., local accuracy and consistency) under the chosen background distribution. As with LIME, image explanations usually operate on super-pixels or masked regions, and the choice of background/reference distribution can influence the resulting attributions.

2.3 Positioning of this study

Prior work has established the conceptual differences between gradient-based visualizations and model-agnostic perturbation methods; however, practitioners still face a practical question: which method is most appropriate for a given diagnostic task under real constraints. We therefore adopt a comparative framing and evaluate Grad-CAM, LIME, and KernelSHAP using (i) a quantitative faithfulness proxy (deletion score) and (ii) runtime per explanation, complemented by qualitative galleries on correct and incorrect predictions. This combination aims to support method selection for both rapid inspection and offline reliability audits.

3 Methodology

This section details the methodological choices of the project: (i) why we selected a VGG-16 style classifier as the reference "black box", (ii) how we instantiated three complementary families of post-hoc explainers (Grad-CAM, LIME, and KernelSHAP), and (iii) which evaluation metrics we used to compare explanations in a way that is both practical and aligned with the goals of trust and debugging.

3.1 Reference classifier and design rationale

We adopt a VGG-16 style CNN as a reference model because it is a well-understood baseline in vision: its layered convolutional structure makes it representative of many production CNN pipelines and provides natural internal feature maps for gradient-based explanations. Since CIFAR-10 images are small (32×32), the architecture is adapted to avoid overly aggressive early downsampling. In practice, this means keeping the early convolutional blocks sufficiently "dense" before pooling so that spatial information is not lost too quickly.

Batch Normalization is inserted after convolutions (Conv–BN–ReLU) to improve training stability and to reduce sensitivity to initialization and learning-rate settings. This choice is particularly relevant for our study because unstable training can confound explainability comparisons: if the classifier itself is poorly calibrated or inconsistent, differences between explanation methods become harder to interpret.

3.2 Explainability methods: what each method is used for

We compare three methods that reflect different assumptions about access to the model and different intended use cases.

Grad-CAM (gradient-based, fast inspection). Grad-CAM leverages gradients flowing into a chosen convolutional layer to produce a coarse localization map of regions that support the predicted class [1]. The key methodological choice is the explanation layer: we use the last convolutional block because it typically encodes high-level semantic patterns (e.g., object parts) while preserving a usable spatial grid. This makes Grad-CAM a good candidate for rapid, qualitative sanity checks during development and for interactive diagnosis.

LIME (local surrogate, model-agnostic). LIME explains a single prediction by sampling perturbed inputs around the instance and fitting an interpretable surrogate model that approximates the classifier locally [2]. For images, interpretability depends on the notion of "features"; we therefore use super-pixels rather than raw pixels so that the explanation highlights contiguous regions. Methodologically, this choice trades spatial precision for interpretability: the explanation components are easier to reason about, but the results depend on the segmentation granularity and on the perturbation distribution.

KernelSHAP (axiomatic attribution, model-agnostic). SHAP methods aim to provide feature attributions with strong theoretical properties, approximating Shapley values from cooperative game theory [3]. As with LIME, we operate on super-pixels (or masked regions) to keep the attributions human-interpretable. KernelSHAP additionally requires a background (reference) distribution to define what it means for a region to be "absent"; we use a representative background set of 100 CIFAR-10 samples to balance faithfulness and computational cost.

3.3 Evaluation criteria and the choice of metrics

A central goal of this project is to go beyond subjective visual quality and to compare explainers using measurable criteria that matter for real usage.

Faithfulness ("does the explanation reflect the model's true evidence?"). We use a deletion-based faithfulness metric: pixels (or regions) are progressively removed in order of decreasing importance according to the explanation, and we track how quickly the model's confidence for the target class drops. We chose deletion because it directly tests the causal impact of highlighted regions on the model output, and it can be applied consistently across Grad-CAM, LIME, and SHAP even though they produce different types of attributions. In addition, deletion is intuitive for practitioners: if an explainer marks a region as critical, removing it should significantly harm the prediction.

Efficiency ("is the method usable in practice?"). We report average runtime per explanation in seconds. This metric reflects a practical constraint: gradient-based methods typically require only one backward pass, while perturbation-based methods may require hundreds of model evaluations. Runtime therefore helps position each method in a realistic workflow (interactive debugging vs. offline auditing).

Qualitative protocol (successes vs. failures). Finally, we complement quantitative metrics with curated visual galleries for correctly and incorrectly classified samples. The methodological motivation is that some failure modes are not captured by a single scalar: explanations may look plausible while focusing on spurious textures, or methods may disagree strongly on the same input. Inspecting successes and failures side-by-side allows us to interpret deletion scores and runtime results in the context of real model behavior.

4 Experiments

This section describes the experimental setup in sufficient detail to enable reproducibility: dataset preparation, training configuration for the classifier, the exact instantiation of each explanation method, and the evaluation metrics.

4.1 Dataset: CIFAR-10

We use the CIFAR-10 image classification dataset, consisting of 60,000 color images of size 32×32 across 10 classes. We follow the standard split provided by the dataset: 50,000 training images and 10,000 test images.

Preprocessing. All images are converted to tensors and normalized channel-wise using mean= $[0.4914, 0.4822, 0.4465]$ and std= $[0.2023, 0.1994, 0.2010]$.

Data augmentation. During training only, we apply standard augmentation to improve generalization: RandomCrop(32, padding=4) and RandomHorizontalFlip. No augmentation is applied at test time.

4.2 Model and training configuration

Architecture. The classifier is a VGG-16 model adapted for CIFAR-10, with 3×3 convolutions (padding=1), interleaved max-pooling, and Batch Normalization in the feature extractor. The classifier head consists of AdaptiveAvgPool2d((1,1)), Flatten, two fully-connected layers of size 4096 with ReLU and Dropout, and a final linear layer to 10 classes.

Optimization. We train for 50 epochs with CrossEntropyLoss and stochastic gradient descent (SGD) using learning rate 0.1, momentum 0.9, and weight decay 5×10^{-4}. The batch size is 128. We use a cosine annealing learning-rate schedule (CosineAnnealingLR) with $T_{\max} = 50$.

Compute environment. All experiments are run with CUDA on a NVIDIA GeForce RTX 3060 GPU.

4.3 Explainability methods: implementation settings

For each explained image, we generate a normalized heatmap in $[0, 1]$ for each method.

Grad-CAM. Grad-CAM explanations are computed from gradients of the target class score with respect to feature maps of a chosen convolutional layer; gradients are pooled to obtain channel weights, and the weighted feature-map combination is passed through a ReLU and normalized.

LIME. We use `lime.lime_image.LimeImageExplainer` with `num_samples`=1000, `top_labels`=10, and `hide_color`=0. The explanation mask is constructed from the returned super-pixel segments and normalized to $[0, 1]$.

SHAP. We use `shap.GradientExplainer` initialized with a background dataset (subset of the training set) to approximate expectations. SHAP values are computed for each input image, aggregated by summing absolute attributions across color channels, and normalized to $[0, 1]$.

4.4 Evaluation protocol and metrics

Classification performance. We report overall accuracy on the CIFAR-10 test set. We also visualize the confusion matrix and training curves (loss/accuracy) for transparency and debugging.

Faithfulness: deletion score (AUC). To evaluate whether an explanation reflects evidence truly used by the classifier, we compute a deletion score. For each image, pixels are ranked by importance according to the heatmap (highest to lowest). We then iteratively mask the top-ranked pixels in 10 steps (replacing them with the dataset mean) and record the model probability for the target class after each masking step. The deletion score is the area under the resulting probability-vs.-step curve (AUC). Lower AUC indicates more faithful explanations (confidence drops faster when "important" pixels are removed).

Efficiency: runtime. We measure average wall-clock runtime per explanation (seconds/image) for Grad-CAM, LIME, and SHAP using the same hardware and software stack.

5 Results

This section reports (i) classification performance of the VGG-16 model and (ii) a comparative evaluation of Grad-CAM, LIME, and SHAP using runtime and deletion score, complemented by qualitative visual examples.

5.1 Model performance

The adapted VGG-16 model achieves a best validation accuracy of 92.79% on CIFAR-10. Figure 1 shows the learning dynamics over the 50 training epochs, and Figure 2 reports class-wise confusions.

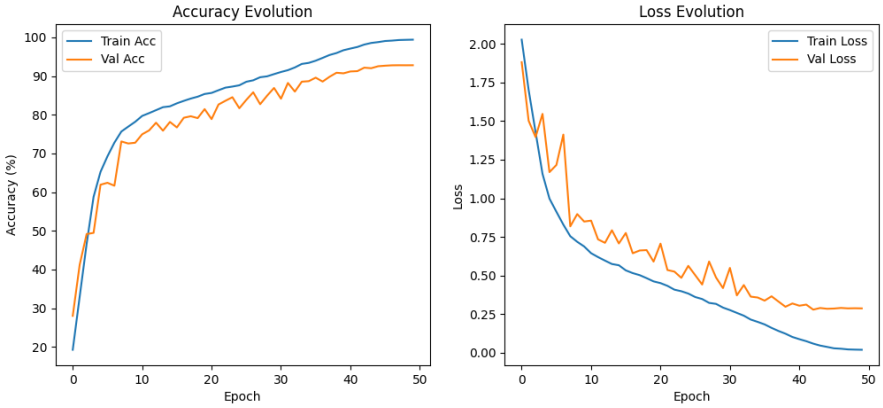

Figure 1: Training history (accuracy/loss) for the VGG-16 classifier.

In addition to overall accuracy, Table 1 summarizes per-class precision/recall/F1. The largest performance drops occur on visually ambiguous animal categories (cat/dog), which is consistent with common CIFAR-10 failure modes.

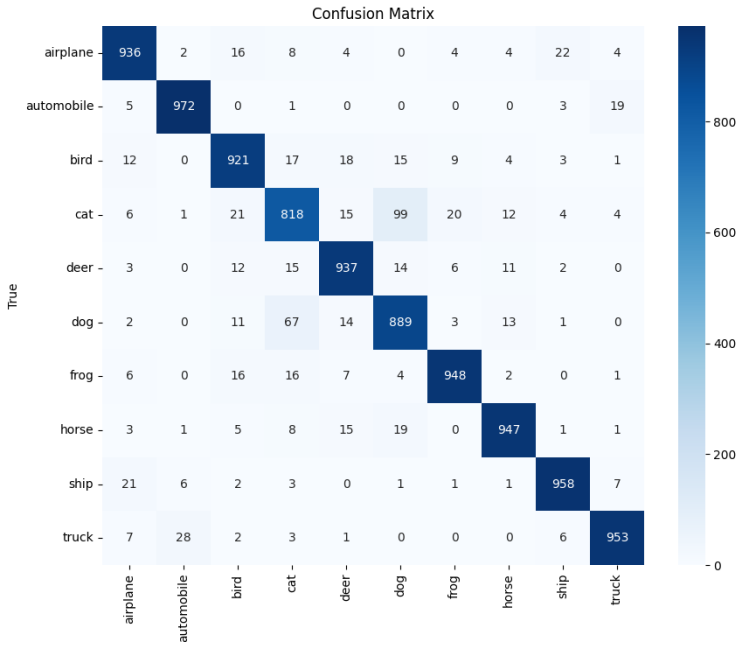

Figure 2: Confusion matrix on the CIFAR-10 test set.

5.2 Quantitative comparison of explainability methods

We compare Grad-CAM, LIME, and SHAP along two axes: efficiency (runtime per explanation) and faithfulness (deletion score AUC; lower is better). Figure 3 visualizes the comparison, while Table 2 provides the exact averages.

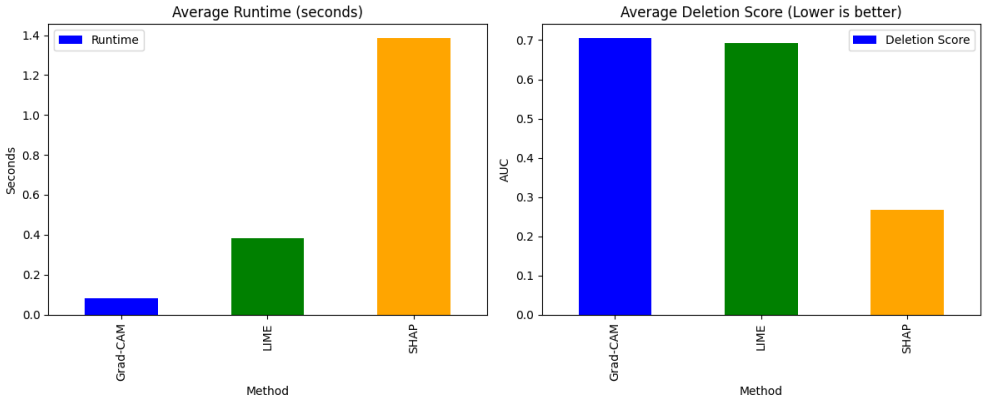

Figure 3: Runtime and deletion-score comparison for Grad-CAM, LIME, and SHAP.

Overall, Grad-CAM is the fastest method, making it suitable for interactive inspection. LIME is moderately slower due to perturbation sampling but produces super-pixel explanations that can be easier to communicate to non-experts. SHAP (GradientExplainer) is the most computationally expensive, yet it achieves the lowest deletion AUC in our setup, indicating that its highlighted pixels are more causally aligned with the model's output.

Table 1: Classification report on the CIFAR-10 test set (10,000 images; 1,000 per class).

Class	Precision	Recall	F1-score	Support
airplane	0.94	0.94	0.94	1000
automobile	0.96	0.97	0.97	1000
bird	0.92	0.92	0.92	1000
cat	0.86	0.82	0.84	1000
deer	0.93	0.94	0.93	1000
dog	0.85	0.89	0.87	1000
frog	0.96	0.95	0.95	1000
horse	0.95	0.95	0.95	1000
ship	0.96	0.96	0.96	1000
truck	0.96	0.95	0.96	1000
Macro avg	0.93	0.93	0.93	10000
Weighted avg	0.93	0.93	0.93	10000

Table 2: Average runtime and deletion score (AUC) for each explanation method. Lower deletion AUC indicates higher faithfulness.

Method	Avg runtime (s)	Avg deletion score (AUC)
Grad-CAM	0.083	0.705
LIME	0.384	0.692
SHAP	1.386	0.267

5.3 Qualitative visual comparisons

To complement the quantitative metrics, we present side-by-side visualizations for representative correct and incorrect predictions (Figures 4 and 5). These examples help interpret when methods agree on salient regions and when they diverge, which is particularly useful for diagnosing spurious correlations.

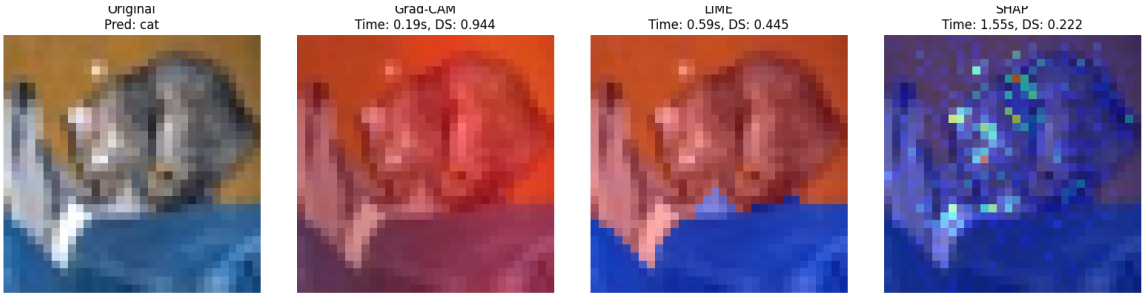

Figure 4: Example of side-by-side explanations for a correctly classified sample.

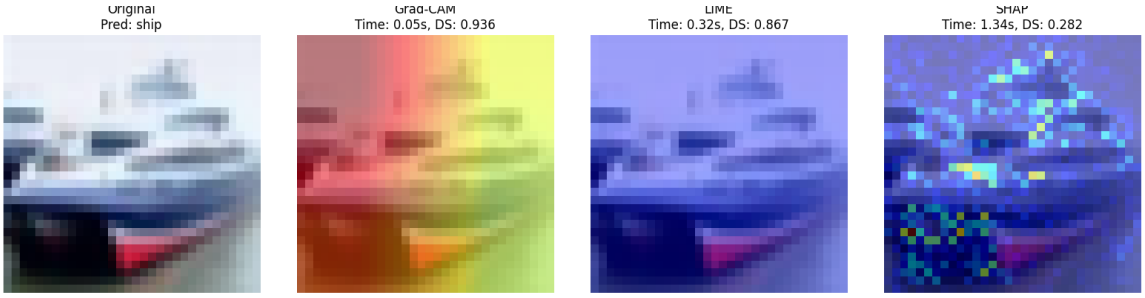

Figure 5: Example of side-by-side explanations for an incorrectly classified sample.

6 Discussion

Our results highlight a recurring theme in explainable AI for vision: there is no universally "best" explanation method, and practical selection depends on the intended use case and on constraints such as time, access to gradients, and the level of detail required.

6.1 Interpreting the accuracy results

The classifier reaches 93.00% test accuracy, with the most challenging categories being visually similar animals (notably cat and dog). This pattern is expected on CIFAR-10 and is relevant for explainability: ambiguous categories are precisely where users are most likely to request explanations, yet explanations are also more likely to expose reliance on texture cues or background context.

6.2 Speed–faithfulness trade-off across explainers

The quantitative comparison suggests a clear efficiency–faithfulness trade-off. Grad-CAM is the fastest method (0.083s per image on our setup), making it appropriate for interactive debugging sessions where many examples must be inspected quickly. However, its deletion score is relatively high, indicating that the highlighted regions are not always the most causally critical for the prediction.

LIME occupies a middle ground in runtime (0.384s) but shows deletion scores comparable to Grad-CAM in this experiment. One interpretation is that, under our segmentation and sampling settings,

LIME produces explanations that are visually interpretable but not substantially more faithful in the deletion sense. In practice, LIME can still be valuable because its super-pixel components map to discrete regions that are easier to communicate to non-technical stakeholders.

SHAP (GradientExplainer) is the slowest method (1.386s), but it achieves the lowest deletion AUC, suggesting that its attributions more strongly align with features that the model relies on. This makes SHAP well suited for offline reliability audits where compute cost is acceptable and the goal is to stress-test the model's evidence.

6.3 When methods disagree

Qualitative examples confirm that methods can disagree even on the same input. Disagreement can be caused by (i) the resolution of explanations (Grad-CAM is coarse by design), (ii) the feature definition (super-pixels for LIME/SHAP), and (iii) different sensitivities to masking strategies and background references. Importantly, disagreement is not necessarily a defect: it can serve as a diagnostic signal that the model may be using multiple cues (e.g., object shape versus texture) or that the decision boundary is fragile.

6.4 Practical implications: recommended workflow

Based on the observed trade-offs, we recommend a layered workflow:

- **Fast triage:** use Grad-CAM to rapidly screen many samples and identify suspicious behaviors (e.g., consistent attention to corners/backgrounds).
- **Case-level investigation:** use LIME to produce region-based explanations that can be discussed and validated with humans, especially when communicating results.
- **Offline auditing:** use SHAP to evaluate faithfulness more rigorously and to focus on high-impact cases (misclassifications, high-confidence errors, or critical classes).

6.5 Limitations

This study has several limitations. First, faithfulness is evaluated with a deletion protocol that depends on a specific masking choice (replacement with dataset mean) and a fixed number of steps (10); alternative perturbations could change absolute scores. Second, LIME and SHAP results depend on segmentation granularity and the background/reference distribution, respectively. Finally, our conclusions are drawn on CIFAR-10 and a VGG-style CNN; outcomes may differ for larger datasets or different architectures.

7 Conclusion

This report studied post-hoc explainability for deep CNN image classification by comparing Grad-CAM, LIME, and SHAP on a VGG-16 model trained on CIFAR-10. Beyond producing visually

appealing heatmaps, we emphasized practical evaluation through two complementary lenses: (i) faithfulness via a deletion-based AUC score and (ii) usability via runtime.

Our experiments show that the VGG-16 classifier achieves strong predictive performance (93.00% test accuracy). In terms of explainability, we observe a consistent trade-off: Grad-CAM is highly efficient and well suited to interactive inspection, while SHAP provides more faithful attributions under our deletion protocol at substantially higher computational cost. LIME offers an interpretable region-based compromise, although its faithfulness in our setup remains close to Grad-CAM.

A key takeaway is that quantitative evaluation is essential: qualitative heatmaps alone can be misleading, whereas deletion-based testing provides a more actionable proxy for whether highlighted regions truly matter to the model. From a deployment perspective, this supports a layered approach in which Grad-CAM is used for fast triage and SHAP for offline auditing of high-impact cases.

Future work. Several extensions could strengthen the generality of these findings. First, the study could be repeated with modern architectures such as Vision Transformers, and with higher-resolution datasets where spatial explanations behave differently. Second, additional faithfulness and stability metrics (e.g., insertion tests, sensitivity to perturbation noise, or agreement across random seeds) would provide a more complete picture of explanation reliability. Finally, explainability could be integrated into a mitigation loop, where explanations are used to detect spurious correlations and guide data augmentation or debiasing strategies.

References

[1] R. R. Selvaraju et al. *Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization*. ICCV, 2017.

[2] M. T. Ribeiro, S. Singh, and C. Guestrin. *"Why Should I Trust You?" Explaining the Predictions of Any Classifier*. KDD, 2016.

[3] S. M. Lundberg and S.-I. Lee. *A Unified Approach to Interpreting Model Predictions*. NeurIPS, 2017.