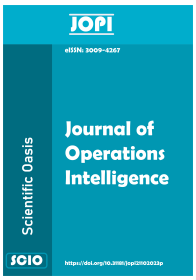
SCIENTIFIC OASIS

Journal of Operations Intelligence

Journal homepage: www.jopi-journal.org
ISSN: 3009-4267



When Pixels Lie: The Hidden Toll of Concept Drift

Gurgen Hovakimyan[1,2,*], Jorge Miguel Bravo[1,2]

[1] NOVA IMS-Information Management School, New University of Lisbon, 1070-312, Lisbon, Portugal
[2] Centro de Investigação em Gestão de Informação (MagIC), 1070-312, Lisbon, Portugal

ARTICLE INFO

Article history:
Received 5 June 2026
Received in revised form 10 August 2026
Accepted 17 September 2026
Available online 21 September 2026

Keywords:
Concept drift; Image classification; Model explainability; Grad-cam; Statistical drift detection

ABSTRACT

Concept drift poses a major challenge to the reliability and transparency of computer vision systems. This study provides a systematic analysis of how four drift types, such as noise, blur, rotation, and brightness, affect both model performance and Grad-CAM/Grad-CAM++ explainability across CIFAR-10 and Tiny ImageNet datasets using three CNN architectures (ResNet-18, DenseNet-121, ShuffleNet v2). Following recent robustness literature, we treat these controlled test-time transformations as *synthetic covariate-shift* proxies for the input-distribution component of drift rather than as temporal concept drift in the strict sense. For CIFAR-10, accuracy declines from a baseline of 79-86% to as low as 24% under strong noise drift, while SSIM and IoU between baseline and drifted heatmaps fall from nearly 100% to as low as 18% and 25%, respectively. On Tiny ImageNet the degradation is even more pronounced: top-1 accuracy collapses from 40-60% to below 4% under the strongest noise drift, with SSIM and IoU dropping to as low as 10% (per-architecture values are reported in Section 4). Rotation and brightness prove markedly milder than noise and blur, indicating that high-frequency corruptions are the principal threat to both accuracy and attribution stability. Image-level paired tests (Wilcoxon signed-rank and paired t-test) with effect sizes and bootstrap confidence intervals, complemented by pixel-level distributional diagnostics (KS, Anderson-Darling, Wilcoxon rank-sum), confirm significant shifts in heatmap distributions, revealing that drift alters not only model outputs but also internal attribution patterns. The results demonstrate that different drift mechanisms degrade interpretability in distinct ways and that explainability metrics are often more sensitive to drift than accuracy itself. This work provides, to the best of our knowledge, one of the first unified, quantitative evaluations of explainability degradation under controlled visual drift, highlighting the need for adaptive, drift-aware XAI pipelines in dynamic real-world environments.

1. Introduction

In real-world applications, machine learning models often encounter data distributions that differ from those on which they were trained, a phenomenon known as concept drift [1]. Concept drift occurs when the statistical properties of input data or their relationships with output labels change over time [2]. This issue is particularly pronounced in high-stakes domains such as healthcare, semiconductor manufacturing, agriculture, recommendation systems, and fraud detection, where subtle distribu-

* *Corresponding author.*
E-mail address: 20231150@novaims.unl.pt

https://doi.org/10.31181/jopi41202670

tional shifts can significantly degrade model performance [3]. Recent works in medical imaging and precision agriculture demonstrate that even robust architectures-such as DenseNet variants, YOLO-based detectors, or hybrid CNN-BiLSTM pipelines-can experience severe performance drops when exposed to real-world acquisition variability [4]. These studies highlight that domain changes, lighting variations, sensor noise, and environmental fluctuations act as natural sources of drift, affecting both prediction reliability and downstream decision-making.

The urgency of this issue has only accelerated in recent years, as distribution shift remains a major failure mode when computer vision systems transition from static training environments to production. Furthermore, Garcia-Rubio *et al.* [5] demonstrated how concept drift severely deteriorates bounding-box accuracy in critical object detection models like YOLO, requiring dynamic AI-human collaborative monitoring. Similarly, recent overarching frameworks linking distribution shift directly to AI safety underscore that failing to handle non-stationary visual data represents a fundamental vulnerability in trustworthy AI [6].

In the context of image classification, concept drift may manifest through environmental changes, lighting variations, sensor noise, or transformations such as rotation and blur. These changes can significantly degrade model performance, with accuracy dropping as models struggle to generalise to new distributions [7].

To address these challenges, researchers increasingly rely on explainable AI (XAI) methods, which provide insights into how models make predictions [8]. Grad-CAM and its extensions (e.g., Grad-CAM++ and Smooth Grad-CAM++) produce visual heatmaps that identify regions influential for classification decisions [9, 10]. However, despite their popularity, the reliability of Grad-CAM-based explanations under drifted conditions remains poorly understood [11]. Subtle or significant distributional shifts can produce misleading heatmaps, complicating interpretation and potentially undermining trust in model decisions [12].

In parallel, statistical methods such as the Kolmogorov-Smirnov test, Anderson-Darling test, Wilcoxon test, and t-test are frequently used to detect distributional changes in evolving data streams [13].

1.1 Key Contributions and Identified Gaps

This study systematically examines the impact of concept drift on both model performance and interpretability, bridging a crucial gap in existing research. While prior studies primarily focus on accuracy degradation, little attention has been paid to how drift affects explainability methods such as Grad-CAM. Our findings reveal that concept drift can distort heatmaps, leading to misleading interpretations, an issue that has remained largely unexplored. Furthermore, although statistical methods are widely used to detect distributional shifts, their application in assessing changes in model explanations remains limited. By integrating statistical tests with explainability analysis, this study provides a novel perspective on concept drift evaluation.

1.2 Methodological Approach and Empirical Strategy

To investigate the interplay between concept drift, model performance, and interpretability, this study examines four types of drift-*noise, blur, rotation,* and *brightness*-at varying intensities. We train three convolutional neural network architectures, ResNet-18 [14], DenseNet-121 [15], and ShuffleNet v2 [16], on the CIFAR-10 and Tiny ImageNet datasets and introduce controlled drift perturbations to the test set. Model performance is assessed by classification accuracy, and interpretability is evaluated using Grad-CAM and Grad-CAM++ heatmaps. The impact of drift is quantified through complementary similarity metrics SSIM, IoU (evaluated at several thresholds), and continuous measures (Pearson correlation, cosine similarity, and MSE) reported with bootstrap 95% confidence intervals. Statistical significance is assessed primarily at the image level using paired tests (the Wilcoxon signed-rank test and the paired t-test) accompanied by Cohen's d effect sizes, and is complemented by pixel-level distributional diagnostics (the Kolmogorov-Smirnov, Anderson-Darling, and Wilcoxon rank-sum tests). This integrated approach enables a comprehensive understanding of drift effects on deep learning models.

1.3 Research Questions

To address these identified gaps, this study investigates the following research questions:

- **RQ1:** How do different types of concept drift (e.g., noise, blur, rotation, and brightness) affect the performance of image classification models?
- **RQ2:** How does concept drift impact the reliability and interpretability of Grad-CAM and Grad-CAM++ based explainability in image classification tasks?
- **RQ3:** Can statistical methods highlight significant changes in Grad-CAM and Grad-CAM++ heatmaps caused by concept drift?

The remainder of this paper is organised as follows. Section 2 reviews related work on concept drift, explainability techniques, and statistical approaches to drift detection. Section 3 details the experimental setup, including the dataset, drift types, evaluation metrics, and statistical tests. Section 4 presents the experimental findings and discusses their implications for the performance and explainability of the model. Section 5 discusses the main limitations of the study. Finally, section 6 concludes the paper and outlines the directions for future research.

2. Related Work

2.1 Concept Drift and Statistical Methods for Detection

Concept drift occurs when the statistical properties of data change over time, which can severely degrade machine learning performance. This issue is common in dynamic fields like medicine, agriculture, and security, where variables such as lighting, sensors, or seasons naturally fluctuate. As deep learning models are increasingly deployed in these evolving environments, robust mechanisms to detect and adapt to drift are essential.

Concept drift can be categorised by its nature. Virtual drift refers to changes in the input distribution without altering the decision boundary, while novel class drift arises from the emergence of previously unseen classes in the data. Real drift, on the other hand, describes changes in the conditional distribution of labels given the inputs. Figure 1 illustrates these distinctions.

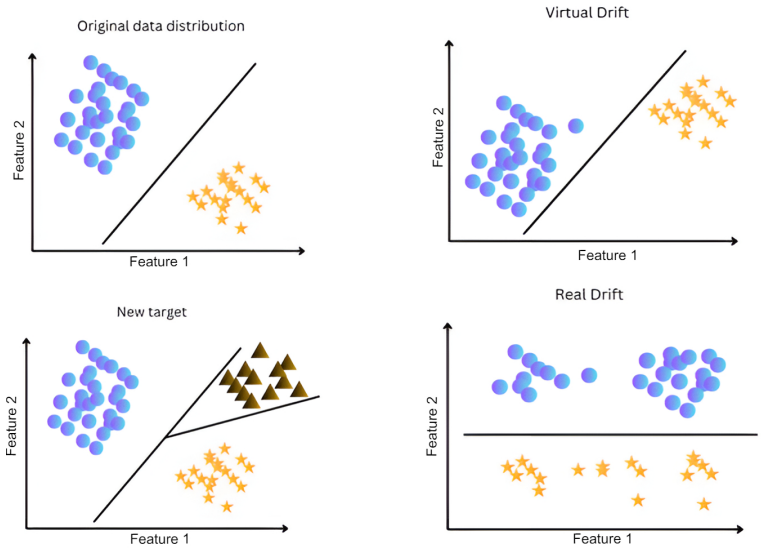


Fig. 1. Types of concept drift based on data distribution: virtual drift, novel class drift, and real drift [17]

Concept drift can also be classified based on its temporal behaviour. Sudden drift represents an abrupt change in the data distribution, whereas incremental drift involves a stepwise progression over time. Reoccurring drift describes distribution changes that revert to previously observed patterns, while gradual drift refers to smooth and continuous changes in the distribution. Figure 2 provides a visual representation of these temporal categories.

Detecting concept drift is crucial for ensuring the reliability of deployed models. Various statistical methods have been proposed for this purpose. For instance, the Drift Detection Method (DDM)

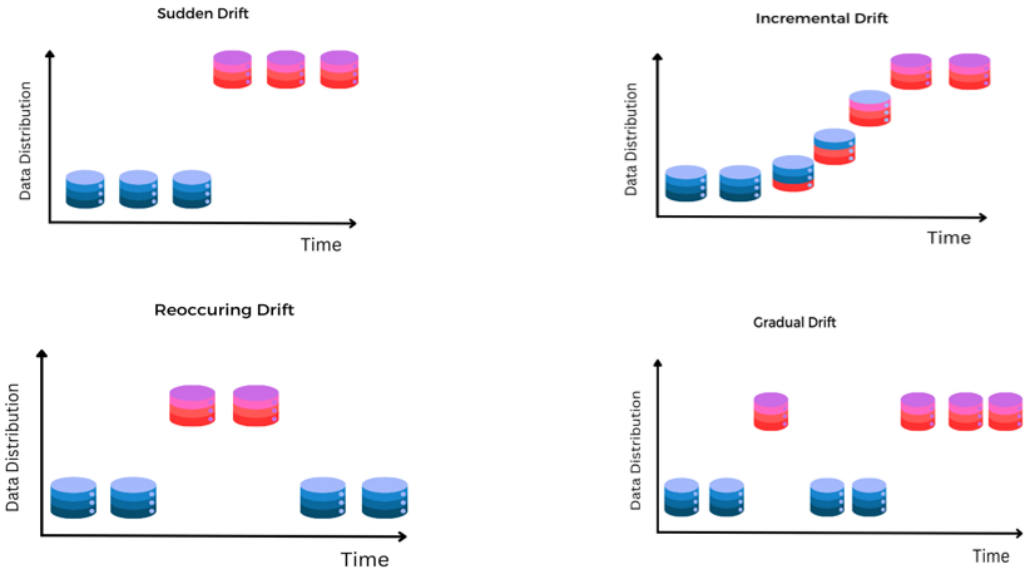


Fig. 2. Types of concept drift based on time: sudden, incremental, reoccurring, and gradual drift [17]

monitors the online error rate of classifiers to detect changes in the data distribution [18]. Adaptive Windowing (ADWIN) dynamically adjusts the size of a sliding window to detect changes in the data stream, ensuring that the window contains data from a stationary distribution [19]. Additionally, the Kolmogorov-Smirnov Windowing (KSWIN) method utilises the Kolmogorov-Smirnov test to detect changes in the distribution of incoming data by comparing the most recent data with a reference window [20].

Recognizing the limitations of traditional statistical methods on high-dimensional data, recent research has pivoted toward deep learning representations. Greco *et al.* [21] proposed DriftLens, a framework that performs unsupervised concept drift detection in real-time by analysing distribution distances directly within the latent space of unstructured data models. For a broader perspective on distribution shift, Wiles *et al.* [22] provided a fine-grained analysis of how model performance varies across shifted data distributions. Geirhos *et al.* [23] further showed that ImageNet-trained CNNs are biased toward texture rather than shape, which helps explain their sensitivity to visual changes. Additionally, unified paradigms for evolving machine learning are increasingly treating drift detection, continual learning, and catastrophic forgetting as interconnected challenges in non-stationary environments [24]. Recent approaches have also been proposed to handle heterogeneous multi-class drift dynamically [25].

Drift detection methods, such as DDM and ADWIN, and more recent models developed based on these approaches, are effective at identifying distributional changes in data streams. However, these methods primarily focus on identifying drift and provide limited insight into how these shifts affect a model's interpretability. This limitation is particularly challenging in real-world scenarios, where understanding the interplay between drift and model behaviour is crucial for building trust in machine learning systems.

2.2 Explainability Techniques for Image Classification

As deep learning models [26] become more complex, explainability has become central to ensuring transparency and user trust. For Convolutional Neural Networks (CNNs) [27] deployed in high-stakes environments like medical imaging, understanding exactly how a model reaches its conclusions is essential for safe and reliable deployment. Explainability methods have been developed to provide insights into these complex models, enhancing transparency and trust. A widely used technique is Gradient-weighted Class Activation Mapping (Grad-CAM), which generates visual explanations by highlighting the regions in an input image that are most influential for a model's prediction. Grad-CAM achieves this by utilising the gradients of a target class flowing into the final convolutional layer

to produce a localisation map of important regions [8]. Despite its effectiveness, Grad-CAM has certain limitations, such as difficulty in accurately localising multiple occurrences of the same class within an image and challenges in capturing fine-grained details. To address these issues, Grad-CAM++ was introduced as an extension that provides better visual explanations by considering higher-order derivatives, resulting in improved object localisation and the ability to explain multiple object instances in a single image [9]. To further improve these visualisation techniques, Smooth Grad-CAM++ combines the strengths of SMOOTHGRAD [28] and Grad-CAM++ to produce clearer and more detailed visual explanations. By adding noise to the input image and averaging the resulting Grad-CAM++ maps, Smooth Grad-CAM++ reduces visual noise and improves the clarity of the generated explanations [10]. These explainability methods are crucial for interpreting deep learning models, especially in sensitive applications such as medical imaging, where understanding the rationale behind a model's decision is vital to ensure reliability and trustworthiness [29]. However, the robustness of these explainability techniques under dynamic conditions, such as those initiated by concept drift, remains largely unexplored. Most studies assume a static data distribution, overlooking potential degradations in explainability metrics like SSIM and IoU when the data distribution shifts. This raises concerns about the reliability of explainability methods in evolving environments.

The vulnerability of XAI under evolving conditions has become a focal point of recent research. Teixera *et al.* [30] formally identified that concept drift creates major challenges for explainable AI, as changing operational distributions silently alter feature importance, decision pathways, and the model's structural reasoning processes. This degradation limits the utility of XAI in dynamic perception tasks, such as 3D scene understanding and autonomous driving, where trustworthy monitoring interfaces must accurately reflect real-time uncertainty under shift [31]. Moreover, researchers are now attempting to invert the relationship, using XAI not just for visualisation, but as an active mechanism for feature selection and adaptation to shifting data [32]. Despite these acknowledgments in the recent literature, quantitative empirical studies measuring exactly how specific image drifts corrupt XAI spatial overlap (e.g., IoU and SSIM) remain scarce.

2.3 Positioning with Respect to Existing Work

To the best of our knowledge, no prior work simultaneously (1) introduces controlled pixel-level drifts (noise, blur, rotation, brightness) on image-classification benchmarks such as CIFAR-10 and Tiny ImageNet, (2) evaluates multiple CNN architectures under identical drift conditions, (3) quantifies the effect of drift on Grad-CAM and Grad-CAM++ using similarity metrics such as SSIM and IoU, and (4) applies formal statistical hypothesis tests directly to saliency-map distributions. This combination of controlled perturbation, explainability analysis, and statistical validation appears to be unexplored in the existing literature.

The closest conceptual work to ours is the analysis by Adebayo *et al.*, who demonstrate through "Sanity Checks for Saliency Maps" that gradient-based explanations can behave unreliably and may fail basic consistency criteria under certain perturbations [12]. While their study provides foundational insight into the fragility of saliency methods, it does not frame the problem in terms of concept drift, nor does it analyse heatmap similarity or distributional changes across controlled drift intensities.

Research on distribution shift in computer vision has largely focused on changes in data domains rather than explanation stability. For example, Michaelis *et al.* introduce the *Robustness* benchmark to study natural shifts such as fog, snow, and noise [33], and Recht *et al.* evaluate distribution shift using new test sets for CIFAR and ImageNet [34]. These works examine accuracy degradation but do not assess how explainability outputs behave under shift.

Consequently, our contribution is complementary to all existing directions: rather than proposing a new drift detector or explainability method, we present, to the best of our knowledge and based on the literature surveyed above, one of the first systematic, quantitative characterisations of how specific, controlled drift mechanisms distort Grad-CAM/Grad-CAM++ explanations and model perfor-

mance across multiple CNN architectures.

3. Methodology

3.1 Data Preparation

The experiments in this study are conducted using two image classification benchmarks: CIFAR-10 and Tiny ImageNet. CIFAR-10 consists of 60,000 RGB images of spatial resolution 32×32 distributed across 10 object classes, with 50,000 images used for training and 10,000 for testing [35]. All images are normalised using a channel-wise mean of $(0.5, 0.5, 0.5)$ and a standard deviation of $(0.5, 0.5, 0.5)$, and no data augmentation is applied in order to isolate the effects of controlled concept drift perturbations on model behaviour. Data loading is performed using PyTorch's `DataLoader` [36] with a batch size of 128 to ensure efficient shuffling and batching. To further assess the scalability and robustness of the proposed analysis, the Tiny ImageNet dataset is additionally employed as a higher-complexity benchmark. Tiny ImageNet contains 200 object classes, each with 500 training images and 50 validation images, resulting in a total of 100,000 training samples and 10,000 validation samples. All images are RGB with a spatial resolution of 64×64, and the same normalisation principles are applied unless otherwise stated. A summary of the datasets used in this study is provided in Table 1.

All experiments were conducted on a workstation equipped with a NVIDIA GeForce RTX 3080 Laptop GPU (12 GB VRAM), an Intel Core i7 CPU with 8 cores, and 32 GB DDR5 RAM (4800 MHz). All training and evaluation procedures were executed using Python 3.12.5 with CUDA 11.8 and PyTorch 2.5.0.

Table 1
Summary of datasets used in this study

Dataset	Image Size	Classes	Train Size	Test/Val Size
CIFAR-10	32×32	10	50,000	10,000
Tiny ImageNet	64×64	200	100,000	10,000

3.2 Model Definition and Training

The classification model used in this study is a modified ResNet-18 architecture [14], a widely recognised convolutional neural network for image classification. ResNet-18 was selected due to its well-documented robustness in handling complex image features and its widespread use as a benchmark in concept drift research. Its residual connections help mitigate the vanishing gradient problem, ensuring stable performance even under shifting data distributions, making it an appropriate choice for evaluating concept drift effects. To ensure that our findings are not architecture-dependent and to enhance the robustness of our conclusions, we also include two additional models: DenseNet-121 [15] and ShuffleNet v2 [16]. DenseNet-121, known for its densely connected layers, facilitates gradient flow and feature reuse, making it an effective choice for examining how drift impacts feature representation. ShuffleNet v2, a lightweight model designed for efficient computation, provides insights into how concept drift affects resource-constrained architectures. The inclusion of these diverse architectures allows for a comprehensive evaluation of drift effects across different model designs. The models are implemented using PyTorch and initialised from ImageNet-pretrained backbones, with the classification head re-initialised for the target number of classes; no data augmentation is applied so that the observed changes are attributable to the induced drift rather than to augmentation. Training is performed using the cross-entropy loss function and the Adam optimiser. Adam (Adaptive Moment Estimation) is chosen due to its efficiency in handling non-stationary objectives, a crucial property for studying drifted data. It combines the advantages of AdaGrad and RMSProp, dynamically adapting the learning rates for each parameter [37]. AdaGrad scales the learning rate inversely proportional to the square root of past gradients, making it effective for sparse updates, while RMSProp stabilises training by using a moving average of squared gradients [38].

We use Adam, which maintains running averages of both the gradients ($\mathbf{m}_t$) and their squared

values ($\mathbf{v}_t$). These averages are then used to update the model weights as follows:

$$\mathbf{m}_t = \beta_1 \mathbf{m}_{t-1} + (1-\beta_1)\mathbf{g}_t, \tag{1}$$

$$\mathbf{v}_t = \beta_2 \mathbf{v}_{t-1} + (1-\beta_2)\mathbf{g}_t^2, \tag{2}$$

$$\hat{\mathbf{m}}_t = \frac{\mathbf{m}_t}{1-\beta_1^t}, \quad \hat{\mathbf{v}}_t = \frac{\mathbf{v}_t}{1-\beta_2^t}, \tag{3}$$

$$\theta_t = \theta_{t-1} - \eta \frac{\hat{\mathbf{m}}_t}{\sqrt{\hat{\mathbf{v}}_t} + \epsilon}. \tag{4}$$

Definitions: $\mathbf{g}_t$ is the gradient of the loss function at time step t. β_1, β_2 are the exponential decay rates for the moment estimates. η is the learning rate. ϵ is a small constant to avoid division by zero.

To ensure the reproducibility and clarity of the experimental setup, all hyperparameters and configuration details used throughout training, drift simulation, and explainability analysis are explicitly summarised. The chosen settings reflect a balance between computational efficiency and empirical robustness, allowing the models to converge reliably while keeping the experimental pipeline tractable across multiple architectures and drift scenarios. Standardised preprocessing, consistent training schedules, and uniform drift intensities were applied across all models to enable fair comparison. Table 2 provides a complete overview of the parameters used in the study, including training hyperparameters, model configurations, drift settings, evaluation metrics, and statistical test specifications.

Table 2
Summary of all experimental parameters used in the study

Training and Model Parameters	
Learning rate	0.001
Batch size	128
Optimizer	Adam
Loss function	Cross-Entropy
Epochs	Up to 30, early stopping (patience 7); best epoch (≈5) checkpointed
Models	ResNet-18, DenseNet-121, ShuffleNet v2
Pretrained weights	ImageNet (classification head re-initialised)
Input size	32×32 (CIFAR-10), 64×64 (Tiny ImageNet)
Drift Simulation Parameters	
Drift types	Noise, Blur, Rotation, Brightness
Drift intensities γ	0–0.5 (12 levels: 10^{-6}–10^{-3}, 0.01–0.5)
Noise drift	Additive Gaussian noise $x + \mathcal{N}(0, \gamma^2)$
Blur drift	Gaussian blur; kernel $k = 3 + \lfloor 10\gamma \rfloor$, $k \in \{3, 5, 7, 9\}$
Brightness drift	`ColorJitter(brightness=`γ`)`
Rotation drift	`RandomRotation` with angle $\theta = 45^\circ\gamma$
Explainability & Evaluation Parameters	
Explainability methods	Grad-CAM, Grad-CAM++
Similarity metrics	SSIM; IoU (thresholds 0.3/0.5/0.7); Pearson; cosine; MSE
Statistical tests	Image-level paired (Wilcoxon signed-rank, paired t) + Cohen's d, bootstrap 95% CI; pixel-level KS, AD, Welch t, Wilcoxon rank-sum; $\alpha = 0.05$
Target layer (Grad-CAM)	Penultimate conv block (ResNet `layer3`, DenseNet `denseblock3`, ShuffleNet `stage3`)

To address potential concerns regarding the short effective training duration, an empirical convergence analysis was conducted. Because the models are fine-tuned from ImageNet-pretrained backbones without data augmentation to preserve the purity of the drift evaluation, they converge rapidly. As illustrated by the representative learning curve for ResNet-18 on CIFAR-10 in Figure 3, validation loss reaches its global minimum at epoch 5. Training uses a budget of up to 30 epochs with early stopping (patience 7), and the best (epoch-5) snapshot is checkpointed; extending training beyond this point consistently triggers overfitting across the evaluated architectures, evidenced by climbing validation loss and plateauing validation accuracy. The early-stopped, best-epoch model therefore

captures robust, generalizable feature representations before over-indexing on the static training distribution.

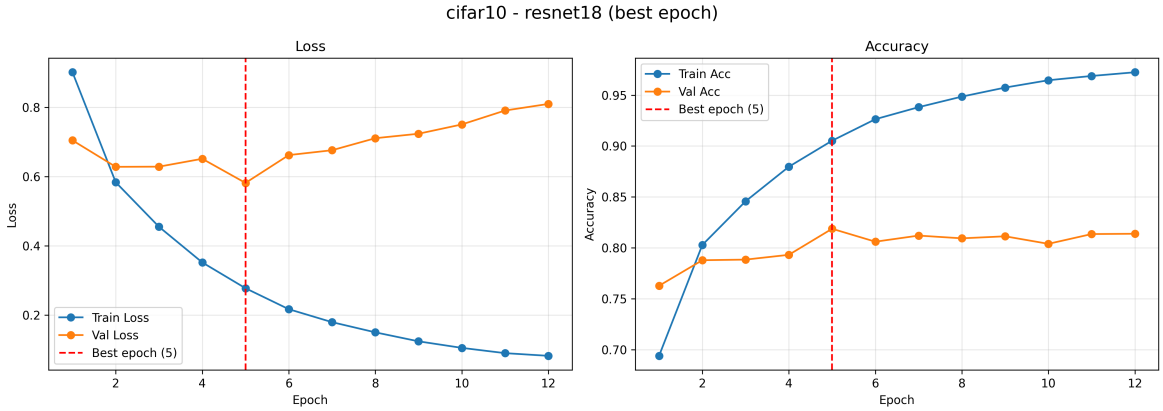


Fig. 3. Representative training and validation loss/accuracy curves (ResNet-18 on CIFAR-10), demonstrating optimal convergence and the onset of overfitting around epoch 5

3.3 Drift Simulation

In the concept-drift literature a distinction is drawn between *covariate shift* (a change in the input distribution $P(x)$ with the labelling function $P(y \mid x)$ held fixed), *real concept drift* (a change in $P(y \mid x)$), and *synthetic corruption* (controlled, parametric image transformations used to probe robustness, as in the ImageNet-C/CIFAR-C benchmarks). The four operators studied here noise, blur, brightness, and rotation are applied to a static test set and therefore do not alter the ground-truth labels; they induce a synthetic, parametrically controlled *covariate shift* rather than temporal concept drift in an evolving data stream. We adopt these operators deliberately, because their intensity can be dialled continuously, allowing us to isolate the effect of a single, well-defined distributional change on both accuracy and attribution maps. Throughout the paper we therefore use "drift" as shorthand for this controlled input-distribution shift, and we restrict our conclusions accordingly (Section 5).

To analyse the impact of concept drift on model performance and explainability, four types of drift are simulated: noise, blur, brightness, and rotation. Each drift type is applied to the test data at varying intensities, ranging from minimal to severe. The specific transformations used are as follows:

- **Noise**: Gaussian noise scaled by intensity is added to the pixel values.
- **Blur**: Gaussian blur is applied with a kernel size proportional to the intensity.
- **Brightness**: Image brightness is adjusted linearly based on intensity.
- **Rotation**: Images are rotated by an angle proportional to the intensity, with a maximum of 45 degrees.

These transformations are implemented dynamically during evaluation, ensuring that the drifted datasets reflect realistic scenarios of data distribution changes.

3.4 Grad-CAM and Grad-CAM++ Explainability

We evaluate explainability using Grad-CAM and Grad-CAM++, which generate heatmaps highlighting the regions of an input image most influential to the model's prediction. For every image, the class activation map is computed with respect to the model's *predicted* class (the arg-max of the logits), not the ground-truth label, because at deployment time labels are unavailable and the quantity of practical interest is the explanation actually shown to a user. The same predicted-class convention is used for the clean and drifted versions of each image, so that any change in the heatmap reflects the drift rather than a change of target class. Heatmaps are generated for both the original and drifted test datasets, allowing a direct comparison of explainability under varying drift conditions. Each raw map is min-max normalised to the $[0, 1]$ range on a per-image basis (the standard `pytorch-grad-cam` output), which is what makes the SSIM, IoU and correlation comparisons commensurable across images and thresholds. The maps are extracted from each network's penultimate convolutional stage (ResNet-

18 `layer3`, DenseNet-121 `denseblock3`, ShuffleNet v2 `stage3`): on the low-resolution inputs used here the final stage collapses to a 1×1 spatial map, for which Grad-CAM is degenerate, whereas the penultimate stage retains a 2×2–4×4 map that yields informative saliency.

Two further design choices follow from this setup: first, misclassified images are *retained*: they receive a heatmap for their (possibly wrong) predicted class, since excluding them would bias the analysis towards the easy examples and would confound accuracy loss with attribution change. Second, no spatial resizing of heatmaps is required, because within each dataset all images share a fixed resolution (32×32 for CIFAR-10 and 64×64 for Tiny ImageNet) and the paired clean/drifted maps are therefore already aligned pixel-for-pixel. The per-image, paired quantitative metrics reported below never average maps across different images; the *class-averaged* heatmaps are used only for qualitative visualisation, where we note that averaging over objects appearing at heterogeneous spatial locations can blur fine localisation and is interpreted with corresponding caution. The generated heatmaps are stored for quantitative and qualitative analysis.

3.5 Metrics for Evaluation

To assess the impact of concept drift on model performance and explainability, we use a combination of predictive accuracy, similarity-based metrics, and statistical tests. Accuracy quantifies changes in classification performance, while SSIM and IoU measure structural and spatial alignment of XAI heatmaps before and after drift. In addition, the Mean Squared Error (MSE) captures pixel-wise deviations between attribution maps.

Choice of similarity metrics. SSIM and IoU are used because they capture two complementary and interpretable notions of "the same explanation". SSIM operates directly on the continuous, min-max-normalised attribution field and summarises whether the local luminance/contrast structure of the saliency map is preserved, without requiring any thresholding. IoU, by contrast, targets the practically relevant question of whether the *most salient region* the part of the map a practitioner would actually act on, still overlaps after drift; it is therefore computed on the top-activation mask rather than as a segmentation score. Because attribution maps are continuous fields rather than segmentation masks, we do not rely on IoU alone: we additionally report continuous, threshold-free measures (the Pearson correlation and cosine similarity between the raw maps, alongside MSE) so that conclusions do not hinge on any single binarisation.

IoU threshold and its sensitivity. The IoU mask is obtained by thresholding the per-image min-max-normalised map; we adopt 0.5 (i.e. the upper half of each map's dynamic range) as the default. Because a single threshold could bias the conclusions, we perform a sensitivity analysis and recompute IoU at 0.3, 0.5 and 0.7. The relative ordering of drift types and the monotonic decline of overlap with drift intensity are preserved across all three thresholds (the absolute IoU shifts up at lower thresholds and down at higher ones, as expected), confirming that the reported trends are not artefacts of the 0.5 choice.

Statistical unit and test selection. The statistical unit of analysis is the *image*: each test image contributes one clean heatmap and one drifted heatmap, forming a natural matched pair. Our primary analysis is therefore *paired* at the image level for each image we summarise the clean-versus-drifted change by its per-image similarity and test, across the N images, whether the median change differs from zero using the Wilcoxon signed-rank test, with the paired t-test as a parametric cross-check. Statistical significance is reported together with a paired Cohen's d effect size and a percentile bootstrap 95% confidence interval for the mean similarity, so that practical significance is not inferred from p-values alone. For completeness we also retain the pixel-level, two-sample diagnostics used in earlier drafts (the Kolmogorov-Smirnov, Anderson-Darling and Wilcoxon rank-sum tests) that compare the marginal distributions of clean and drifted heatmap activations; because these pool spatially autocorrelated pixels and treat them as independent, they are highly sensitive and are interpreted only as distributional descriptors, not as the primary evidence.

For brevity, only high-level descriptions of the metrics are provided here. All mathematical definitions and complete formulas are moved to 7.

3.6 Experimental Workflow

The experimental workflow, as presented in the diagram below, comprises the following steps:

1. **Model Training**: Train the CNN model on the CIFAR-10/Tiny ImageNet dataset to establish a baseline for accuracy and explainability metrics.
2. **Baseline Metrics Computation**: Generate XAI heatmaps on the original dataset and compute baseline metrics, including SSIM, IoU, and accuracy.
3. **Drift Simulation**: Apply different types of drift (e.g., noise, blur, brightness, and rotation) at varying intensities to the test dataset.
4. **Drifted Metrics Computation**: Generate XAI heatmaps on drifted datasets and compute corresponding metrics to quantify the effect of drift.
5. **Metric Comparison**: Compare metrics from baseline and drifted datasets to assess the impact of concept drift on explainability and performance.
6. **Statistical Testing**: Perform statistical tests (e.g., KS, AD, t-test, Wilcoxon) to identify significant differences in heatmaps and metrics caused by drift.
7. **Result Analysis**: Visualise average heatmaps and analyse class-specific patterns to facilitate qualitative and quantitative insights.

This workflow, as illustrated in Figure 4, enables a structured and comprehensive evaluation of the impact of concept drift on both model performance and XAI-based explainability.

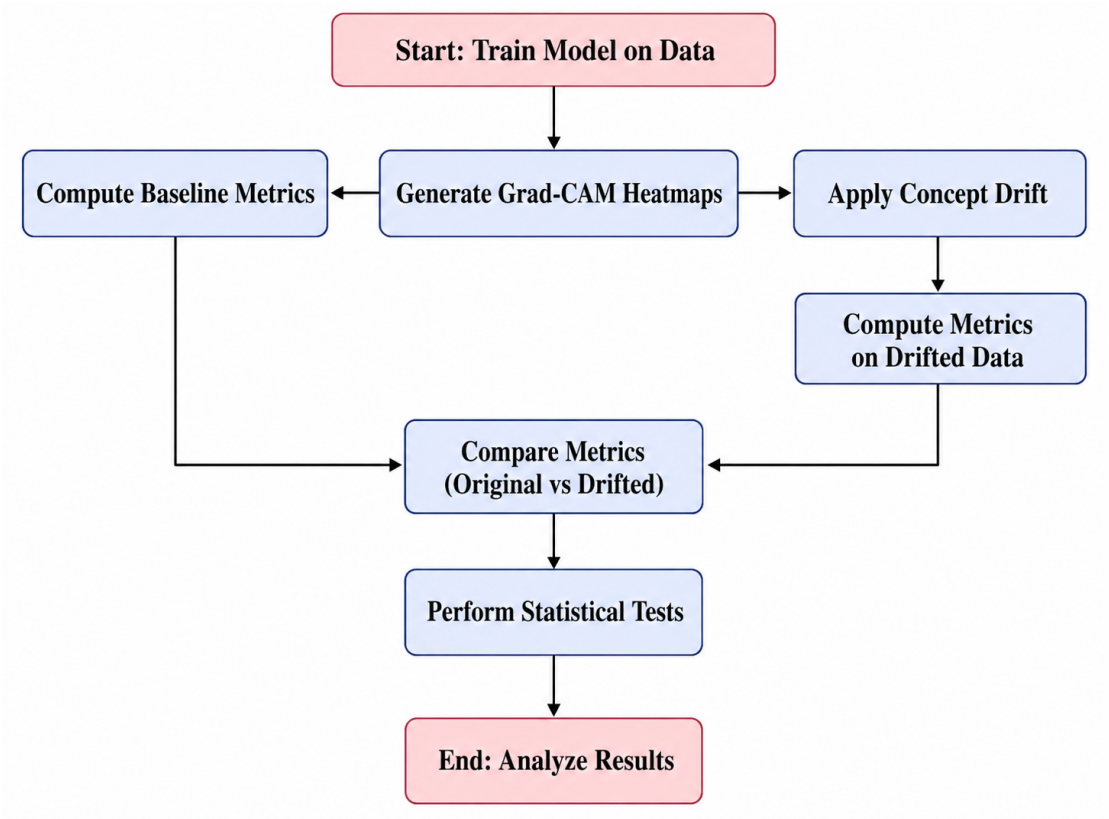


Fig. 4. Experimental Workflow for Evaluating Concept Drift and Explainability

To complement the workflow diagram, the experimental procedure is summarised in Algorithm 1.

Input: Labeled dataset $\mathcal{D}$, ImageNet-pretrained CNN backbone, drift types $\mathcal{T}$, and drift intensities $\mathcal{A}$.
Output: Baseline and drifted performance metrics and explainability degradation analysis.

1. Preprocess dataset $\mathcal{D}$ and split it into training and test sets.
2. Load the ImageNet-pretrained CNN backbone and replace its final layer to match the number of classes.
3. Train the model using cross-entropy loss and the chosen optimiser.
4. Evaluate the model on the clean test set, record accuracy, generate XAI heatmaps, and compute class-wise averages.
5. For each $d \in \mathcal{T}$ and $\alpha \in \mathcal{A}$, apply drift (d, α), evaluate accuracy, generate and average heatmaps, compare clean and drifted maps using SSIM and IoU, and perform the stated statistical tests.
6. Summarise and visualise all performance and explainability metrics.

Algorithm 1: Experimental Pipeline for Drift Analysis

4. Results and Discussion

In this section, we present the experimental results assessing the impact of various types of concept drift (noise, blur, rotation, brightness) on the performance and explainability of the trained image classification model. The results are analysed in terms of accuracy, structural similarity (SSIM), intersection over union (IoU), and statistical significance. Key findings are discussed to provide insights into the interplay between model performance and explainability under drifted conditions.

4.1 Ablation Study

To isolate the contribution of individual drift factors, we conduct an ablation study in which each drift type (noise, blur, brightness, rotation) is applied independently while keeping all other conditions unchanged. The case of zero drift intensity serves as the control condition, representing the model's baseline behavior without any distributional shift. Comparing this baseline to results obtained under increasing drift intensities allows us to quantify the specific impact of each drift component on both accuracy and XAI based explainability.

Across all architectures, noise and blur produce the largest declines in both accuracy and heatmap similarity, whereas brightness and rotation lead to more moderate but consistent degradations. Rotation exhibits a mildly non-linear profile, with minimal disruption at low intensities. These results confirm that different drift mechanisms influence model predictions and attribution maps in distinct ways, highlighting the necessity of evaluating drift sources separately.

4.2 Model Performance and Drift Impact

The baseline (non-drifted) accuracy of the trained ResNet-18 model on CIFAR-10 was 84.06%. Grad-CAM-based explainability metrics, namely the Structural Similarity Index (SSIM) and Intersection over Union (IoU), were computed for heatmaps generated on the test set, and under non-drifted conditions both recorded perfect or near-perfect scores for all four drift types. Blur is the exception at low intensity: its similarity drops as soon as the Gaussian kernel exceeds a single pixel, so it degrades earlier than the other drifts as intensity increases.

When concept drift was introduced, the model performance and explainability metrics declined significantly, with the extent of the decline varying by drift type and intensity. Tables 3, 4, 5, and 6 provide a detailed quantitative summary of the results for noise, blur, rotation, and brightness drifts, respectively, after the model is trained on ResNet-18.

These results demonstrate the diverse impacts of the four drift types on model performance and explainability. Noise caused the most severe degradation, with accuracy dropping from 84.06% under no drift to 26.41% at the highest intensity, while SSIM and IoU declined to 43.86% and 52.00%, respectively. Blur was similarly damaging, reducing accuracy to 26.88% and SSIM and IoU to 40.67% and 48.98%; blur already lowers SSIM to about 62.61% at the smallest effective kernel, once the blur kernel exceeds a single pixel. In contrast, rotation and brightness were considerably milder for

Table 3
Performance and explainability metrics for noise drift. Used training model ResNet-18

Drift Intensity	Accuracy (%)	Average SSIM (%)	Average IoU (%)
0	84.06	100.00	99.69
0.01	84.22	98.01	96.72
0.05	82.03	89.25	87.67
0.1	78.75	76.57	76.97
0.2	63.44	55.86	60.95
0.3	48.59	47.29	53.95
0.4	37.19	44.54	52.61
0.5	26.41	43.86	52.00

Table 4
Performance and explainability metrics for blur drift. Used training model ResNet-18

Drift Intensity	Accuracy (%)	Average SSIM (%)	Average IoU (%)
0	84.06	100.00	99.69
0.01	67.34	62.61	66.17
0.05	67.34	62.61	66.17
0.1	67.34	62.61	66.17
0.2	48.28	50.50	56.76
0.3	34.06	43.15	50.77
0.4	26.88	40.67	48.98
0.5	26.88	40.67	48.98

Table 5
Performance and explainability metrics for rotation drift. Used training model ResNet-18

Drift Intensity	Accuracy (%)	Average SSIM (%)	Average IoU (%)
0	84.06	100.00	99.69
0.01	84.06	100.00	99.69
0.05	84.22	96.69	95.40
0.1	81.41	86.73	85.80
0.2	81.56	85.75	85.86
0.3	78.75	80.28	81.52
0.4	74.06	67.52	69.90
0.5	70.62	67.58	70.24

Table 6
Performance and explainability metrics for brightness drift. Used training model ResNet-18

Drift Intensity	Accuracy (%)	Average SSIM (%)	Average IoU (%)
0	84.06	100.00	99.69
0.01	84.22	99.09	98.03
0.05	83.75	96.28	94.55
0.1	83.75	94.20	92.20
0.2	81.88	86.95	85.23
0.3	80.62	82.10	81.48
0.4	79.53	78.82	78.36
0.5	79.22	74.83	75.37

this ImageNet-pretrained model: rotation left accuracy at 70.62% with SSIM and IoU of 67.58% and 70.24%, and brightness left accuracy at 79.22% with SSIM and IoU of 74.83% and 75.37%. These findings indicate that high-frequency corruptions (noise and blur) dominate both accuracy loss and attribution instability, whereas geometric and photometric shifts (rotation and brightness) are comparatively benign for pretrained CNNs, emphasising the importance of drift-type-specific detection and mitigation strategies.

Consistent results were obtained for the DenseNet-121 and ShuffleNet v2 models (full tables in the supplementary material). At the strongest drift intensity (0.5):

- DenseNet-121 (baseline accuracy 85.94%):
  - ***Noise*** reduced accuracy to 23.91%, SSIM to 18.10% and IoU to 34.67%.
  - ***Blur*** reduced accuracy to 37.03%, SSIM to 25.30% and IoU to 37.83%.
  - ***Rotation*** reduced accuracy to 74.69%, SSIM to 54.78% and IoU to 60.12%.
  - ***Brightness*** reduced accuracy to 82.34%, SSIM to 69.48% and IoU to 70.69%.
- ShuffleNet v2 (baseline accuracy 79.38%):
  - ***Noise*** reduced accuracy to 24.38%, SSIM to 18.36% and IoU to 24.54%.
  - ***Blur*** reduced accuracy to 29.53%, SSIM to 24.64% and IoU to 34.48%.
  - ***Rotation*** reduced accuracy to 69.53%, SSIM to 57.33% and IoU to 58.97%.
  - ***Brightness*** reduced accuracy to 74.22%, SSIM to 57.70% and IoU to 57.08%.

Both architectures reproduce the CIFAR-10 ordering: noise and blur are the most destructive to accuracy and attribution similarity, while rotation and brightness are markedly milder.

A similar experimental procedure was applied to the Tiny ImageNet dataset to verify whether the observed drift patterns generalise to a more complex and higher-resolution benchmark. Despite the differences in dataset size, resolution, and label granularity, the overall degradation trends closely matched those obtained on CIFAR-10.

For **ResNet-18**, the baseline top-1 accuracy on Tiny ImageNet was 40.16%; under the strongest noise drift it collapsed to 0.94%, with SSIM and IoU falling to 18.89% and 19.85%. For **DenseNet-121** the baseline was 59.84%, falling to 3.12% under noise, with SSIM and IoU dropping to 13.06% and 16.83%. For **ShuffleNet v2** the baseline was 53.75%, falling to 0.31% under noise, with SSIM and IoU dropping to 10.24% and 9.99%. As on CIFAR-10, noise and blur produced the steepest degradation of both accuracy and attribution similarity, while rotation and brightness were milder; every drift, however, was more damaging on Tiny ImageNet than on CIFAR-10, consistent with its finer-grained 200-class label space and lower baseline accuracy. Finally, we directly compared the two attribution methods with an image-level *paired* test a Wilcoxon signed-rank test on the per-image difference in heatmap similarity, with a paired Cohen's d effect size for every architecture, drift type and intensity. Across all datasets and architectures the difference was statistically significant, and Grad-CAM++ consistently retained more structural similarity under drift than Grad-CAM, with medium-to-large effect sizes (d between 0.49 and 1.27 at the strongest intensity). Both methods nevertheless degraded monotonically as drift intensity increased, so although Grad-CAM++ is the more drift-robust visualisation, neither is immune; the dominant driver of attribution instability remains the underlying feature distortion caused by the drift rather than the choice of visualisation technique. The complete per-architecture paired results are reported in the supplementary material (Section 8), and this behaviour is examined further in subsection 4.3.

4.3 Explainability Heatmaps

Figure 5 shows Grad-CAM heatmaps for two representative classes, *Class 8* and *Class 9*, under non-drifted conditions and with the most severe drift (intensity = 0.5) for noise, brightness, rotation, and blur drifts. The top row (*Class 8*) illustrates a scenario where significant visual differences are observed between the original and drifted heatmaps, while the bottom row (*Class 9*) demonstrates a case where the heatmaps remain visually consistent despite the presence of drift.

For *Class 8*, noise and blur drifts resulted in the most substantial changes to the heatmaps. Under noise drift, the heatmap activation regions become diffuse and less focused, indicating that the model struggles to localise key features accurately, and blur similarly disperses the salient region. Rotation and brightness drifts introduce comparatively milder distortions, with the heatmaps retaining more alignment with their original counterparts, consistent with the quantitative ordering reported above.

In contrast, *Class 9* exhibits remarkable robustness across all types of drift. Heatmaps under drifted conditions remain visually similar to the original, with activation regions retaining their focus and structure. This suggests that the interpretability for *Class 9* is less sensitive to the types of drifts analysed, even at high intensities.

These observations highlight the variability in the model response to drift across different classes. While some classes, such as *Class 8*, are highly susceptible to changes in explainability under drift, others, such as *Class 9*, show greater resilience. This underscores the need for class-specific evaluations when assessing the impact of concept drift on model interpretability.

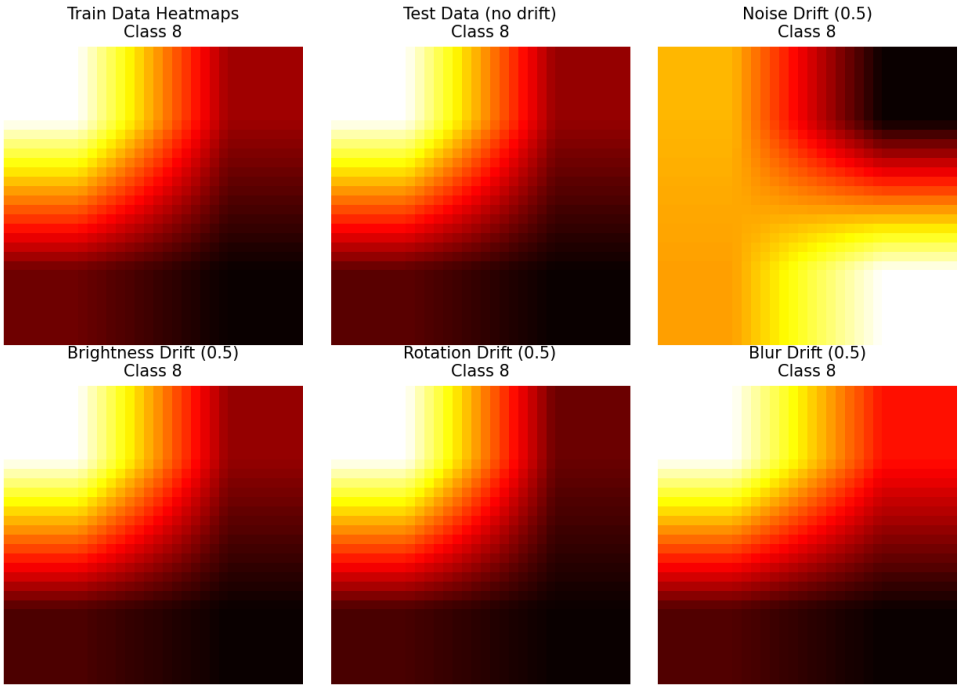
As shown in Figures 5, 6 and 7 similar patterns were found when the model was trained using ResNet-18, DensNet-121 and ShuffleNet v2 architecture. This indicates that the findings are consistent rather than specific to a single architecture, as models with markedly different designs exhibit the same qualitative behaviour; the corresponding full quantitative results for all three architectures are reported in the supplementary material (Section 8).

4.4 Statistical Analysis of Explainability Metrics

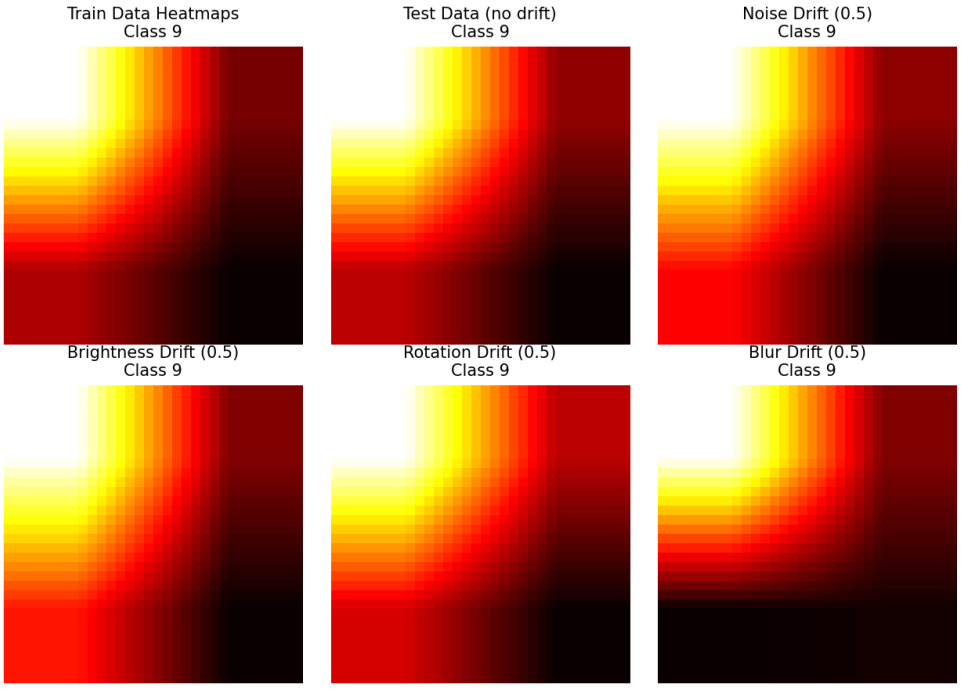
To complement the visual analysis of Grad-CAM heatmaps, statistical tests were applied to quantify the impact of concept drift on explainability metrics. Four statistical tests were employed: the Kolmogorov-Smirnov (KS) test, Anderson-Darling (AD) test, t-test, and Wilcoxon test. These tests compare the distributions of heatmap activations between non-drifted and drifted conditions, providing insight into the significance of observed changes.

Two clarifications on the statistical design are in order. First, the pixel-level, two-sample tests just mentioned (KS, Anderson-Darling, Welch t and Wilcoxon rank-sum) are *distributional* diagnostics: they compare the marginal distributions of clean and drifted activations by pooling pixels, which are spatially autocorrelated and numerous, so these tests are extremely sensitive and their p-values, reported in full in the supplementary material, should be read as distributional descriptors rather than as evidence of practical importance. Second, and as the primary analysis, we adopt the *image* as the statistical unit and compute image-level *paired* tests (Wilcoxon signed-rank and paired t) on the per-image clean-versus-drifted similarity, reporting each result together with a paired Cohen's d effect size and a bootstrap 95% confidence interval, as tabulated in Table 7. This pairing reflects the fact that each image yields a matched clean/drifted heatmap, and the effect sizes ensure that statistically significant differences are also assessed for magnitude. The complete image-level paired results, with effect sizes and confidence intervals for every architecture, drift type and intensity, are provided in the supplementary material (Section 8).

Table 7 reports the primary image-level paired analysis for ResNet-18 on CIFAR-10. For every drift type the Wilcoxon signed-rank test rejects the null of unchanged heatmaps at both intensities, but the paired Cohen's d effect size which measures the magnitude rather than the mere existence of the change tells the substantive story. At the strongest intensity, noise and blur induce large effects ($d = 1.60$ and 1.67) with mean SSIM collapsing to 43.9% and 40.7%, whereas rotation and brightness induce smaller effects ($d = 0.91$ and 0.81) and retain much higher similarity (67.6% and 74.8%). The gap is wider at the intermediate intensity 0.1, where brightness is only a small effect ($d = 0.39$) while noise and blur are already moderate-to-large. This confirms, with effect sizes and confidence intervals rather than p-values alone, that noise and blur are the dominant sources of attribution instability. The same ordering holds for DenseNet-121 and ShuffleNet v2 and on Tiny ImageNet, where the effects are uniformly larger (Cohen's d up to 5.6); the complete per-architecture paired results are provided

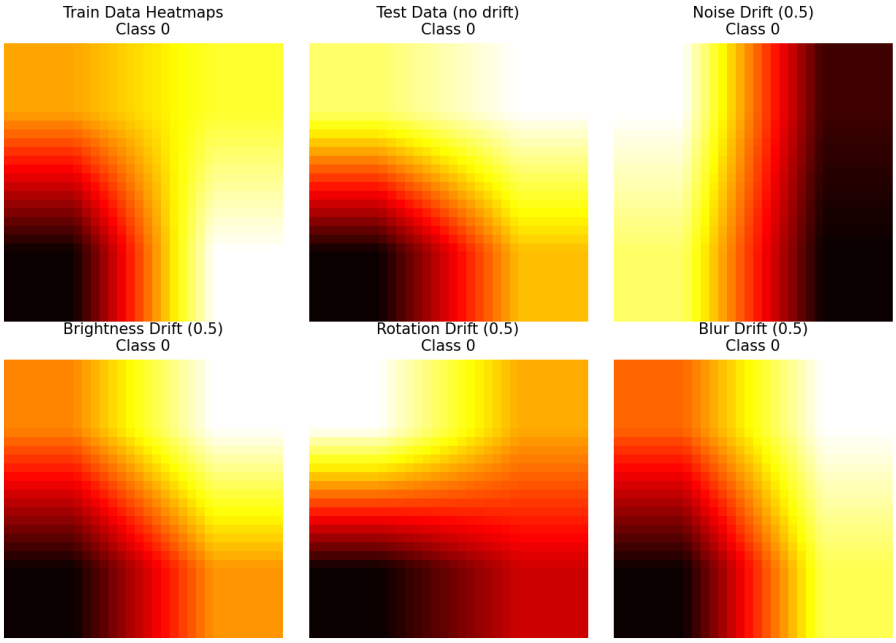


(a) Grad-CAM heatmaps for *Class 8* under non-drifted and severe drift. At intensity = 0.5 the attribution is disrupted most strongly by noise, while brightness, rotation and blur perturb it comparatively little

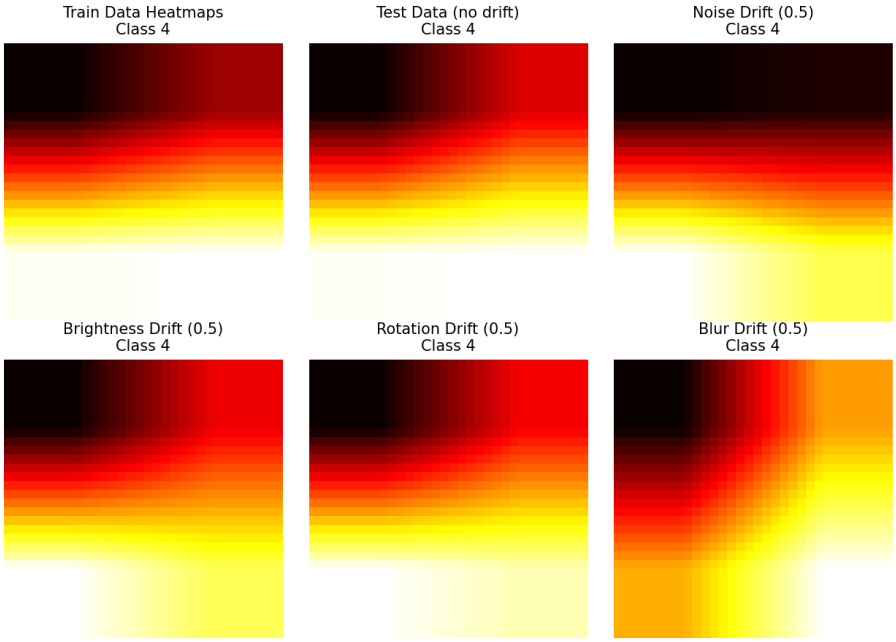


(b) Grad-CAM heatmaps for *Class 9* under non-drifted and severe drift. The activation maps remain more consistent, even with intensity = 0.5

Fig. 5. Comparison of Grad-CAM heatmaps for *Class 8* and *Class 9* under non-drifted and severe drift (intensity = 0.5) conditions across noise, brightness, rotation, and blur

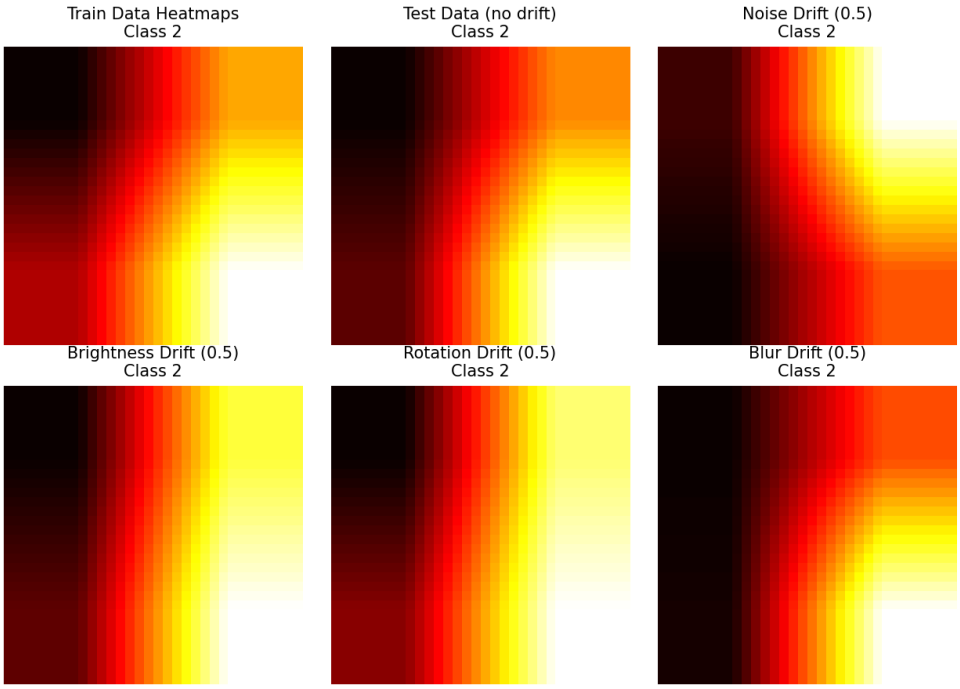


(a) Grad-CAM heatmaps for *Class 0* under non-drifted and severe drift. At intensity = 0.5 the attribution is disrupted most strongly by noise, while brightness, rotation and blur perturb it comparatively little

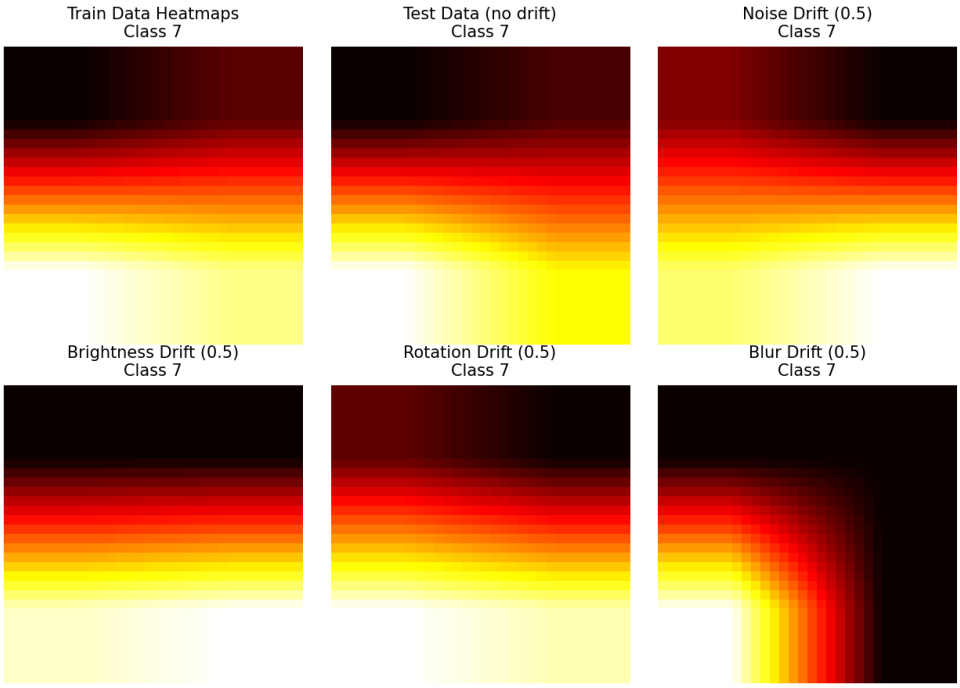


(b) Grad-CAM heatmaps for *Class 4* under non-drifted and severe drift. The activation maps remain largely consistent under noise, brightness and rotation, with blur causing the clearest shift, even at intensity = 0.5

Fig. 6. Comparison of Grad-CAM heatmaps for *Class 0* and *Class 4* under non-drifted and severe drift (intensity = 0.5) conditions across noise, brightness, rotation, and blur



(a) Grad-CAM heatmaps for *Class 2* under non-drifted and severe drift. At intensity = 0.5 the attribution is disrupted most strongly by noise, while brightness, rotation and blur perturb it comparatively little



(b) Grad-CAM heatmaps for *Class 7* under non-drifted and severe drift. The activation maps remain largely consistent, with only a minor change under noise drift, even at intensity = 0.5

Fig. 7. Comparison of Grad-CAM heatmaps for *Class 2* and *Class 7* under non-drifted and severe drift (intensity = 0.5) conditions across noise, brightness, rotation, and blur

in the supplementary material (Section 8).

Table 7
Image-level paired analysis for ResNet-18 on CIFAR-10 (Grad-CAM): mean SSIM with bootstrap 95% CI, Wilcoxon signed-rank p-value, and paired Cohen's d effect size, at intermediate (0.1) and strong (0.5) drift

Drift Type	γ	SSIM (%) [95% CI]	Wilcoxon p	Cohen's d
Noise	0.1	76.6 [74.2, 78.9]	$< 10^{-100}$	0.77
Noise	0.5	43.9 [41.2, 46.6]	$< 10^{-100}$	1.60
Blur	0.1	62.6 [59.8, 65.4]	$< 10^{-100}$	1.03
Blur	0.5	40.7 [37.8, 43.5]	$< 10^{-100}$	1.67
Rotation	0.1	86.7 [84.9, 88.5]	$< 10^{-80}$	0.57
Rotation	0.5	67.6 [64.7, 70.2]	$< 10^{-80}$	0.91
Brightness	0.1	94.2 [93.1, 95.4]	$< 10^{-100}$	0.39
Brightness	0.5	74.8 [72.4, 77.2]	$< 10^{-100}$	0.81

The progressive nature of the effect is evident in Table 7: moving from intensity 0.1 to 0.5 roughly doubles the paired Cohen's d for every drift type, confirming that stronger drift produces both lower heatmap similarity and larger, more certain effect sizes.

4.5 Efficiency Analysis: Runtime and Memory Consumption

In addition to accuracy and explainability evaluation, we quantify the computational cost of the drift-analysis pipeline for each model-dataset combination.

Because the classification backbones are trained once and then reused, we report the cost of the recurring *analysis* stage the full drift sweep (four drift types across twelve intensities), Grad-CAM/Grad-CAM++ generation, similarity computation, and statistical testing rather than the one-time training cost. For each of the three architectures (ResNet-18, DenseNet-121, ShuffleNet v2) and both datasets (CIFAR-10 and Tiny ImageNet) we record the total wall-clock time and the peak GPU memory of this stage; the CPU-memory increase during the stage was negligible (well below 0.3 GB, as the data and model are already resident) and is therefore omitted.

Table 8 summarises these figures, providing a view of the recurring operational cost of drift monitoring and the trade-off between architectural complexity and analysis cost.

Table 8
Drift-analysis runtime and peak GPU memory for each model-dataset configuration (backbones trained once and reused; one-time training cost excluded)

Model	Dataset	Time (min)	Peak GPU (GB)
ResNet-18	CIFAR-10	3.99	0.19
ResNet-18	Tiny ImageNet	7.55	0.44
DenseNet-121	CIFAR-10	3.82	0.44
DenseNet-121	Tiny ImageNet	6.40	1.63
ShuffleNet v2	CIFAR-10	3.29	0.10
ShuffleNet v2	Tiny ImageNet	7.11	0.51

The comparison shows that Tiny ImageNet roughly doubles the analysis time of CIFAR-10 for every architecture, while the time is otherwise similar across architectures because it is dominated by the fixed Grad-CAM sweep. DenseNet-121 has the largest GPU footprint (up to 1.63 GB on Tiny ImageNet), whereas the lightweight ShuffleNet v2 is the most memory-efficient. These figures characterise the recurring cost of drift-aware monitoring rather than one-time training.

5. Limitations

Although this study provides a comprehensive analysis of how pixel-level concept drift affects both model performance and Grad-CAM/Grad-CAM++ explainability, several limitations must be acknowledged. These limitations primarily stem from the controlled nature of the experimental setup, dataset

selection, architectural scope, and the analytical focus of the study.

First, the drift mechanisms examined-noise, blur, rotation, and brightness-are synthetic and intentionally controlled to allow systematic evaluation. While these perturbations represent common sources of visual degradation encountered in real-world systems, natural concept drift can be substantially more complex. In practice, drift may be multi-modal, temporally evolving, or occur in combination across multiple factors, making it more challenging to isolate and analyze. Consequently, although the selected drift types provide valuable insights, they may not capture the full spectrum of distributional shifts encountered in deployed vision systems. Accordingly, we explicitly restrict the empirical conclusions of this study to controlled, synthetic covariate shifts applied to a static test set; we do not claim to characterise temporal concept drift or naturally occurring distribution shift. Extending the analysis to naturally corrupted benchmarks (e.g., CIFAR-10-C and Tiny-ImageNet-C), temporal acquisition variation, domain shift, and combined perturbations is an important direction that we leave to future work.

Second, the experimental evaluation is conducted on CIFAR-10 and Tiny ImageNet, both of which consist of relatively low-resolution images. While these datasets are widely used benchmarks that enable reproducibility and controlled experimentation, they may not fully reflect the visual complexity, semantic richness, and noise characteristics present in real-world applications such as medical imaging, precision agriculture, or industrial inspection. As a result, the magnitude and nature of explainability degradation observed in this study may differ when models are applied to higher-resolution or domain-specific datasets.

Third, the analysis is limited to convolutional neural network architectures, specifically ResNet-18, DenseNet-121, and ShuffleNet v2. Although these models cover a range of architectural complexities and computational profiles, modern vision systems increasingly rely on alternative paradigms such as Vision Transformers or hybrid CNN–Transformer architectures. These models may exhibit different sensitivity to drift and different explainability behaviors, which are not examined in this work.

Finally, the scope of this study is restricted to detecting, quantifying, and analyzing the impact of concept drift on model performance and explainability. Adaptive mitigation strategies-such as continual learning, fine-tuning under drift, or drift-aware training mechanisms-are intentionally excluded to maintain a focused analytical framework. While this allows for a clearer understanding of drift-induced degradation, it limits the ability to assess how the observed effects could be mitigated in practical deployment scenarios.

6. Conclusion and Future Work

This study provided a comprehensive investigation of how pixel-level concept drift affects both predictive performance and explainability in image classification models. By applying four controlled drift types-noise, blur, rotation, and brightness-across multiple intensities, we systematically quantified their impact on model accuracy, Grad-CAM/Grad-CAM++ heatmaps, and structural similarity metrics such as SSIM and IoU. The results demonstrate that concept drift can profoundly alter both the behavior and interpretability of convolutional neural networks, with noise and blur causing the strongest degradation, while rotation and brightness exhibit more moderate but still measurable effects. A complementary battery of statistical tests (KS, AD, Welch t-test, Wilcoxon) confirmed the significance of these changes, particularly at higher drift intensities, and revealed meaningful class-dependent differences in explainability stability.

These findings were validated across three architectures, ResNet-18, DenseNet-121, and ShuffleNet v2, indicating that the observed patterns are not specific to a single model family. The integrated methodology introduced here, combining visual, quantitative, and statistically calibrated evaluation, provides a principled framework for understanding how distributional changes distort both model decisions and their explanations.

For clarity, we delimit what is *directly demonstrated* by our experiments from what constitutes a

broader implication. Directly demonstrated: under controlled, synthetic covariate shift on CIFAR-10 and Tiny ImageNet, increasing drift intensity produces monotonic and statistically significant degradation of Grad-CAM/Grad-CAM++ similarity (SSIM, IoU, and continuous correlation measures), with image-level paired tests and effect sizes confirming the effect; this degradation is consistent across the three CNN architectures and comparable between the two attribution methods; and explainability metrics are frequently more sensitive to drift than accuracy. Broader implications, not directly tested here, are that these effects transfer to naturally occurring or temporal drift, to higher-resolution or domain-specific imagery, and to non-convolutional architectures, and that adaptive mechanisms would mitigate the observed degradation. The latter are hypotheses motivating the future work outlined below rather than results established by this study.

While the study offers important insights, it also highlights several avenues for future work. Expanding the analysis to additional architectures, including Vision Transformers or hybrid CNN-Transformer models, would help determine whether newer designs exhibit improved robustness under drift. Evaluating real-world drift arising from lighting variations, sensor degradation, seasonal shifts, or domain changes would further increase the practical relevance of the approach.

Most importantly, directly addressing the degradation of explainability requires moving beyond detection to active mitigation. Future research must investigate how established adaptation frameworks can be integrated into XAI pipelines under strict resource constraints to stabilise attribution maps. For instance, **continual learning** could allow models to incrementally update their feature extractors in response to incremental drift, ensuring that Grad-CAM heatmaps smoothly transition to newly relevant features without catastrophic forgetting of the original domain. Similarly, **test-time adaptation (TTA)**, which updates model parameters on-the-fly using unlabelled target data, offers a promising avenue to recalibrate internal feature statistics, theoretically restoring the structural focus of heatmaps instantly under sudden shifts. Furthermore, **drift-aware retraining** and regularisation methods could be designed to explicitly penalise erratic changes in XAI maps during model updates, forcing models to maintain consistent feature attributions rather than just accurate predictions. Recent advancements demonstrate that this adaptation must happen dynamically; for instance, hybrid transformer-autoencoder architectures have shown significant promise in improving real-time concept drift detection while maintaining interpretability [39]. However, continuous adaptation introduces significant computational overhead. Solutions like dynamic model update policies that optimise training dynamics while provably limiting update frequency and cost will be essential for deploying these adaptive models on edge devices [40].

In summary, this study establishes that concept drift has a measurable and often severe effect on both model accuracy and interpretability. By providing a rigorous, multi-angle analysis of these effects, it lays the foundation for future research aimed at developing robust, adaptive, and trustworthy models capable of maintaining reliable predictions and explanations in dynamic, real-world environments.

Declaration

During the preparation of this work, the authors used generative AI technologies solely to assist with language translation, grammatical proofreading, and stylistic refinement. After using these tools, the authors thoroughly reviewed and edited the content as needed and take full responsibility for the final integrity and content of the publication.

7. Mathematical Definitions of Metrics

Accuracy

The classification accuracy is computed as:

$$\text{Accuracy} = \frac{\sum_{i=1}^{N} \mathbb{1}(\hat{y}_i = y_i)}{N}, \tag{5}$$

where N is the number of samples.

Structural Similarity Index (SSIM)

$$\mathsf{SSIM}(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}. \tag{6}$$

Intersection over Union (IoU)

$$\mathsf{IoU}(h_1,h_2) = \frac{|h_1 \cap h_2|}{|h_1 \cup h_2|}. \tag{7}$$

Mean Squared Error (MSE)

$$\mathsf{MSE}(h_1,h_2) = \frac{1}{N}\sum_{i=1}^{N}(h_1[i] - h_2[i])^2. \tag{8}$$

Kolmogorov–Smirnov Test

$$D_{KS} = \sup_x |F_1(x) - F_2(x)|. \tag{9}$$

Anderson–Darling Test

$$A^2 = -N - \frac{1}{N}\sum_{i=1}^{N}\left[(2i-1)\ln F(x_i) + (2N+1-2i)\ln(1-F(x_i))\right]. \tag{10}$$

Welch's t-Test

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}. \tag{11}$$

Wilcoxon Rank-Sum Test

$$U_X = R_X - \frac{n_1(n_1+1)}{2}, \tag{12}$$

$$U_X + U_Y = n_1 n_2, \tag{13}$$

For large samples:

$$Z = \frac{U - \mu_U}{\sigma_U}, \quad \mu_U = \frac{n_1 n_2}{2}, \quad \sigma_U = \sqrt{\frac{n_1 n_2 (n_1 + n_2 + 1)}{12}}. \tag{14}$$

Continuous heatmap-similarity measures

For two flattened, min-max-normalised heatmaps a and b, the Pearson correlation and cosine similarity are

$$r(a,b) = \frac{\sum_i (a_i - \bar{a})(b_i - \bar{b})}{\sqrt{\sum_i (a_i - \bar{a})^2}\sqrt{\sum_i (b_i - \bar{b})^2}}, \qquad \cos(a,b) = \frac{a^\top b}{\|a\|\,\|b\|}. \tag{15}$$

Image-level paired analysis

Let $s_i \in [0,1]$ be the per-image similarity between the clean and drifted heatmap of image i and $d_i = 1 - s_i$ the corresponding dissimilarity. The Wilcoxon signed-rank statistic tests $H_0:\ \mathrm{median}(d_i) = 0$ using

$$W = \left|\sum_{i=1}^{N} \mathsf{sgn}(d_i)\, R_i\right|, \tag{16}$$

where R_i is the rank of $|d_i|$. The paired effect size is Cohen's

$$d = \frac{\overline{d_i}}{\mathsf{sd}(d_i)}, \tag{17}$$

and a 95% confidence interval for $\overline{s_i}$ is obtained by the percentile bootstrap over the N images.

8. Supplementary Material

To keep the main text concise while fully substantiating the claims that the findings generalise across architectures and that Grad-CAM and Grad-CAM++ differ in their robustness to drift, the following extended results accompany the paper as machine-readable tables and are reproduced by the released evaluation pipeline.

- **Full per-architecture results.** Complete accuracy, SSIM, IoU, Pearson, cosine and MSE values each with a bootstrap 95% confidence interval for every combination of dataset (CIFAR-10, Tiny ImageNet), architecture (ResNet-18, DenseNet-121, ShuffleNet v2), drift type and the twelve drift intensities. These extend the ResNet-18/CIFAR-10 tables in Section 4.2 to all three architectures and to Tiny ImageNet.
- **Grad-CAM versus Grad-CAM++ paired comparison.** For each architecture, drift type and intensity, a Wilcoxon signed-rank test on the per-image difference in heatmap similarity between the two methods, with a paired Cohen's d effect size, quantifying whether their drift robustness differs.
- **IoU threshold-sensitivity analysis.** IoU recomputed at thresholds 0.3, 0.5 and 0.7, demonstrating that the reported trends are stable with respect to the binarisation threshold.
- **Image-level paired significance.** Wilcoxon signed-rank and paired t-test p-values with Cohen's d effect sizes for the clean-versus-drifted comparison, per architecture and class.

To facilitate future research and ensure the reproducibility of our results, the complete source code, processed datasets, and detailed experimental logs have been made publicly available. All experiments described in Section 4, including the exact random seeds and hyperparameters used, can be replicated using the provided scripts. The official project repository can be accessed at: https://github.com/GurgenHovakimyan/hidden_toll_of_visual_drift

Table 9
Full drift results for DenseNet-121 on CIFAR-10 (Grad-CAM). SSIM shown with bootstrap 95% CI.

Drift	γ	Acc.(%)	SSIM(%) [95% CI]	IoU(%)	Pearson	Cosine
blur	0	85.94	100.00 [100.00, 100.00]	100.00	1.00	1.00
blur	1e-06	73.28	54.11 [51.06, 56.98]	59.53	0.56	0.85
blur	1e-05	73.28	54.11 [51.06, 56.98]	59.53	0.56	0.85
blur	0.0001	73.28	54.11 [51.06, 56.98]	59.53	0.56	0.85
blur	0.001	73.28	54.11 [51.06, 56.98]	59.53	0.56	0.85
blur	0.01	73.28	54.11 [51.06, 56.98]	59.53	0.56	0.85
blur	0.05	73.28	54.11 [51.06, 56.98]	59.53	0.56	0.85
blur	0.1	73.28	54.11 [51.06, 56.98]	59.53	0.56	0.85
blur	0.2	54.84	37.98 [35.00, 40.92]	47.67	0.36	0.77
blur	0.3	43.12	30.13 [27.16, 33.08]	42.21	0.26	0.73
blur	0.4	37.03	25.30 [22.50, 27.99]	37.83	0.19	0.69
blur	0.5	37.03	25.30 [22.50, 27.99]	37.83	0.19	0.69
brightness	0	85.94	100.00 [100.00, 100.00]	100.00	1.00	1.00
brightness	1e-06	85.94	99.99 [99.98, 99.99]	99.89	1.00	1.00
brightness	1e-05	85.94	99.99 [99.98, 100.00]	99.89	1.00	1.00
brightness	0.0001	85.94	99.99 [99.98, 99.99]	99.86	1.00	1.00
brightness	0.001	85.94	99.77 [99.37, 99.97]	99.55	1.00	1.00
brightness	0.01	85.94	99.20 [98.65, 99.62]	98.29	0.99	1.00
brightness	0.05	86.72	96.16 [94.95, 97.20]	94.10	0.97	0.99
brightness	0.1	87.03	92.43 [90.93, 93.95]	89.58	0.94	0.98
brightness	0.2	86.09	86.32 [84.63, 88.14]	83.76	0.90	0.97
brightness	0.3	84.22	80.17 [78.05, 82.36]	78.57	0.85	0.95
brightness	0.4	83.75	75.01 [72.59, 77.51]	75.02	0.80	0.93
brightness	0.5	82.34	69.48 [66.92, 72.26]	70.69	0.75	0.92
noise	0	85.94	100.00 [100.00, 100.00]	100.00	1.00	1.00
noise	1e-06	85.94	99.98 [99.97, 99.99]	99.85	1.00	1.00
noise	1e-05	85.94	99.99 [99.98, 100.00]	99.88	1.00	1.00
noise	0.0001	85.94	99.98 [99.97, 99.99]	99.75	1.00	1.00

Table 9
Continued

Drift	γ	Acc.(%)	SSIM(%) [95% CI]	IoU(%)	Pearson	Cosine
noise	0.001	85.94	99.64 [99.20, 99.87]	99.08	1.00	1.00
noise	0.01	85.47	97.33 [96.30, 98.19]	95.14	0.98	0.99
noise	0.05	83.75	85.19 [83.17, 86.96]	82.26	0.89	0.96
noise	0.1	77.19	65.60 [62.80, 68.45]	67.78	0.69	0.90
noise	0.2	60.47	42.79 [39.71, 45.90]	51.21	0.42	0.79
noise	0.3	41.56	29.22 [26.25, 32.41]	42.66	0.23	0.74
noise	0.4	30.00	22.74 [19.98, 25.63]	37.34	0.14	0.70
noise	0.5	23.91	18.10 [15.32, 21.15]	34.67	0.06	0.67
rotation	0	85.94	100.00 [100.00, 100.00]	100.00	1.00	1.00
rotation	1e-06	85.94	100.00 [100.00, 100.00]	100.00	1.00	1.00
rotation	1e-05	85.94	100.00 [100.00, 100.00]	100.00	1.00	1.00
rotation	0.0001	85.94	100.00 [100.00, 100.00]	100.00	1.00	1.00
rotation	0.001	85.94	100.00 [100.00, 100.00]	100.00	1.00	1.00
rotation	0.01	85.94	100.00 [100.00, 100.00]	100.00	1.00	1.00
rotation	0.05	85.94	100.00 [100.00, 100.00]	100.00	1.00	1.00
rotation	0.1	84.22	81.67 [79.16, 83.94]	82.31	0.85	0.95
rotation	0.2	83.59	71.46 [68.71, 73.96]	73.02	0.75	0.92
rotation	0.3	83.91	74.83 [72.11, 77.48]	75.56	0.79	0.93
rotation	0.4	77.19	62.37 [59.33, 65.26]	66.73	0.64	0.88
rotation	0.5	74.69	54.78 [51.72, 57.80]	60.12	0.57	0.85

Table 10
Full drift results for ResNet-18 on CIFAR-10 (Grad-CAM). SSIM shown with bootstrap 95% CI.

Drift	γ	Acc.(%)	SSIM(%) [95% CI]	IoU(%)	Pearson	Cosine
blur	0	84.06	100.00 [100.00, 100.00]	99.69	1.00	1.00
blur	1e-06	67.34	62.61 [59.75, 65.40]	66.17	0.68	0.87
blur	1e-05	67.34	62.61 [59.75, 65.40]	66.17	0.68	0.87
blur	0.0001	67.34	62.61 [59.75, 65.40]	66.17	0.68	0.87
blur	0.001	67.34	62.61 [59.75, 65.40]	66.17	0.68	0.87
blur	0.01	67.34	62.61 [59.75, 65.40]	66.17	0.68	0.87
blur	0.05	67.34	62.61 [59.75, 65.40]	66.17	0.68	0.87
blur	0.1	67.34	62.61 [59.75, 65.40]	66.17	0.68	0.87
blur	0.2	48.28	50.50 [47.72, 53.20]	56.76	0.55	0.83
blur	0.3	34.06	43.15 [40.35, 45.90]	50.77	0.47	0.80
blur	0.4	26.88	40.67 [37.83, 43.46]	48.98	0.44	0.79
blur	0.5	26.88	40.67 [37.83, 43.46]	48.98	0.44	0.79
brightness	0	84.06	100.00 [100.00, 100.00]	99.69	1.00	1.00
brightness	1e-06	84.06	99.99 [99.97, 100.00]	99.62	1.00	1.00
brightness	1e-05	84.06	99.96 [99.90, 100.00]	99.55	1.00	1.00
brightness	0.0001	84.06	99.85 [99.60, 99.99]	99.42	1.00	1.00
brightness	0.001	84.06	99.93 [99.87, 99.97]	99.38	1.00	1.00
brightness	0.01	84.22	99.09 [98.45, 99.57]	98.03	0.99	0.99
brightness	0.05	83.75	96.28 [95.25, 97.28]	94.55	0.98	0.99
brightness	0.1	83.75	94.20 [93.12, 95.38]	92.20	0.97	0.98
brightness	0.2	81.88	86.95 [85.01, 88.81]	85.23	0.91	0.96
brightness	0.3	80.62	82.10 [80.00, 84.20]	81.48	0.87	0.95
brightness	0.4	79.53	78.82 [76.67, 81.02]	78.36	0.85	0.93
brightness	0.5	79.22	74.83 [72.41, 77.21]	75.37	0.81	0.92
noise	0	84.06	100.00 [100.00, 100.00]	99.69	1.00	1.00
noise	1e-06	84.06	99.95 [99.89, 99.99]	99.58	1.00	1.00
noise	1e-05	84.06	99.98 [99.96, 100.00]	99.59	1.00	1.00
noise	0.0001	84.06	99.97 [99.94, 99.99]	99.56	1.00	1.00
noise	0.001	84.22	99.39 [98.84, 99.82]	98.78	0.99	0.99
noise	0.01	84.22	98.01 [97.19, 98.65]	96.72	0.99	0.99
noise	0.05	82.03	89.25 [87.65, 90.95]	87.67	0.93	0.97
noise	0.1	78.75	76.57 [74.20, 78.94]	76.97	0.82	0.93
noise	0.2	63.44	55.86 [52.95, 58.69]	60.95	0.60	0.84
noise	0.3	48.59	47.29 [44.41, 50.26]	53.95	0.52	0.81
noise	0.4	37.19	44.54 [41.83, 47.35]	52.61	0.49	0.80
noise	0.5	26.41	43.86 [41.16, 46.62]	52.00	0.49	0.80
rotation	0	84.06	100.00 [100.00, 100.00]	99.69	1.00	1.00
rotation	1e-06	84.06	100.00 [100.00, 100.00]	99.69	1.00	1.00

Table 10
Continued

Drift	γ	Acc.(%)	SSIM(%) [95% CI]	IoU(%)	Pearson	Cosine
rotation	1e-05	84.06	100.00 [100.00, 100.00]	99.69	1.00	1.00
rotation	0.0001	84.06	100.00 [100.00, 100.00]	99.69	1.00	1.00
rotation	0.001	84.06	100.00 [100.00, 100.00]	99.69	1.00	1.00
rotation	0.01	84.06	100.00 [100.00, 100.00]	99.69	1.00	1.00
rotation	0.05	84.22	96.69 [95.68, 97.57]	95.40	0.98	0.99
rotation	0.1	81.41	86.73 [84.90, 88.54]	85.80	0.91	0.96
rotation	0.2	81.56	85.75 [83.79, 87.61]	85.86	0.90	0.96
rotation	0.3	78.75	80.28 [77.91, 82.65]	81.52	0.84	0.94
rotation	0.4	74.06	67.52 [64.86, 70.13]	69.90	0.72	0.89
rotation	0.5	70.62	67.58 [64.71, 70.22]	70.24	0.72	0.89

Table 11
Full drift results for ShuffleNet v2 on CIFAR-10 (Grad-CAM). SSIM shown with bootstrap 95% CI.

Drift	γ	Acc.(%)	SSIM(%) [95% CI]	IoU(%)	Pearson	Cosine
blur	0	79.38	100.00 [100.00, 100.00]	95.31	0.95	0.95
blur	1e-06	61.72	45.34 [42.16, 48.50]	50.03	0.45	0.74
blur	1e-05	61.72	45.34 [42.16, 48.50]	50.03	0.45	0.74
blur	0.0001	61.72	45.34 [42.16, 48.50]	50.03	0.45	0.74
blur	0.001	61.72	45.34 [42.16, 48.50]	50.03	0.45	0.74
blur	0.01	61.72	45.34 [42.16, 48.50]	50.03	0.45	0.74
blur	0.05	61.72	45.34 [42.16, 48.50]	50.03	0.45	0.74
blur	0.1	61.72	45.34 [42.16, 48.50]	50.03	0.45	0.74
blur	0.2	43.91	32.93 [29.96, 35.89]	40.76	0.30	0.68
blur	0.3	34.22	27.50 [24.83, 30.32]	37.05	0.22	0.65
blur	0.4	29.53	24.64 [21.94, 27.42]	34.48	0.19	0.62
blur	0.5	29.53	24.64 [21.94, 27.42]	34.48	0.19	0.62
brightness	0	79.38	100.00 [100.00, 100.00]	95.31	0.95	0.95
brightness	1e-06	79.38	99.97 [99.95, 99.99]	95.07	0.95	0.95
brightness	1e-05	79.38	99.93 [99.88, 99.97]	94.92	0.95	0.95
brightness	0.0001	79.38	99.87 [99.81, 99.93]	94.75	0.95	0.95
brightness	0.001	79.38	99.69 [99.54, 99.81]	94.28	0.95	0.95
brightness	0.01	79.84	97.22 [96.43, 97.92]	90.75	0.94	0.95
brightness	0.05	79.69	91.14 [89.79, 92.47]	84.06	0.90	0.93
brightness	0.1	79.38	84.41 [82.63, 86.26]	78.03	0.85	0.91
brightness	0.2	77.66	73.89 [71.55, 76.34]	69.74	0.77	0.87
brightness	0.3	75.94	67.38 [64.82, 70.07]	64.07	0.70	0.84
brightness	0.4	75.00	62.15 [59.52, 64.84]	60.49	0.65	0.81
brightness	0.5	74.22	57.70 [54.82, 60.45]	57.08	0.60	0.79
noise	0	79.38	100.00 [100.00, 100.00]	95.31	0.95	0.95
noise	1e-06	79.38	99.96 [99.92, 99.98]	95.12	0.95	0.95
noise	1e-05	79.38	99.94 [99.89, 99.97]	94.98	0.95	0.95
noise	0.0001	79.38	99.86 [99.79, 99.92]	94.80	0.95	0.95
noise	0.001	79.22	98.66 [97.96, 99.23]	92.80	0.94	0.95
noise	0.01	80.31	93.00 [91.58, 94.33]	86.40	0.91	0.93
noise	0.05	77.97	73.86 [71.38, 76.39]	69.53	0.76	0.87
noise	0.1	70.47	55.96 [52.96, 59.04]	55.64	0.55	0.78
noise	0.2	53.44	36.36 [33.23, 39.39]	39.07	0.32	0.62
noise	0.3	40.47	25.76 [22.74, 28.57]	31.56	0.18	0.54
noise	0.4	29.69	21.86 [19.07, 24.64]	27.91	0.13	0.50
noise	0.5	24.38	18.36 [15.79, 20.99]	24.54	0.10	0.45
rotation	0	79.38	100.00 [100.00, 100.00]	95.31	0.95	0.95
rotation	1e-06	79.38	100.00 [100.00, 100.00]	95.31	0.95	0.95
rotation	1e-05	79.38	100.00 [100.00, 100.00]	95.31	0.95	0.95
rotation	0.0001	79.38	100.00 [100.00, 100.00]	95.31	0.95	0.95
rotation	0.001	79.38	100.00 [100.00, 100.00]	95.31	0.95	0.95
rotation	0.01	79.38	100.00 [100.00, 100.00]	95.31	0.95	0.95
rotation	0.05	79.69	93.71 [92.22, 95.08]	88.93	0.91	0.93
rotation	0.1	78.12	80.03 [77.73, 82.37]	77.09	0.80	0.89
rotation	0.2	75.47	72.42 [69.53, 75.17]	71.99	0.72	0.86
rotation	0.3	70.78	49.93 [46.98, 52.71]	52.95	0.50	0.76
rotation	0.4	67.03	43.47 [40.53, 46.27]	46.36	0.41	0.71
rotation	0.5	69.53	57.33 [54.12, 60.29]	58.97	0.56	0.78

Table 12
Full drift results for DenseNet-121 on Tiny ImageNet (Grad-CAM). SSIM shown with bootstrap 95% CI.

Drift	γ	Acc.(%)	SSIM(%) [95% CI]	IoU(%)	Pearson	Cosine
blur	0	59.84	100.00 [100.00, 100.00]	99.84	1.00	1.00
blur	1e-06	30.94	37.19 [34.96, 39.47]	33.84	0.42	0.73
blur	1e-05	30.94	37.19 [34.96, 39.47]	33.84	0.42	0.73
blur	0.0001	30.94	37.19 [34.96, 39.47]	33.84	0.42	0.73
blur	0.001	30.94	37.19 [34.96, 39.47]	33.84	0.42	0.73
blur	0.01	30.94	37.19 [34.96, 39.47]	33.84	0.42	0.73
blur	0.05	30.94	37.19 [34.96, 39.47]	33.84	0.42	0.73
blur	0.1	30.94	37.19 [34.96, 39.47]	33.84	0.42	0.73
blur	0.2	19.53	28.93 [27.02, 30.94]	27.44	0.29	0.68
blur	0.3	15.16	24.81 [22.91, 26.74]	24.31	0.24	0.65
blur	0.4	11.09	22.89 [21.00, 24.86]	22.72	0.20	0.63
blur	0.5	11.09	22.89 [21.00, 24.86]	22.72	0.20	0.63
brightness	0	59.84	100.00 [100.00, 100.00]	99.84	1.00	1.00
brightness	1e-06	59.84	99.97 [99.95, 99.97]	99.18	1.00	1.00
brightness	1e-05	59.84	99.94 [99.91, 99.96]	99.01	1.00	1.00
brightness	0.0001	59.84	99.78 [99.50, 99.95]	98.84	1.00	1.00
brightness	0.001	59.84	99.71 [99.41, 99.87]	98.33	1.00	1.00
brightness	0.01	59.84	98.19 [97.44, 98.85]	94.64	0.99	0.99
brightness	0.05	58.44	93.38 [92.07, 94.63]	87.35	0.96	0.98
brightness	0.1	57.50	87.97 [86.19, 89.61]	80.76	0.91	0.96
brightness	0.2	55.94	78.13 [75.94, 80.37]	69.78	0.83	0.92
brightness	0.3	54.69	71.92 [69.74, 74.14]	64.17	0.79	0.90
brightness	0.4	53.75	64.34 [62.05, 66.58]	56.80	0.71	0.87
brightness	0.5	52.50	58.33 [56.03, 60.50]	51.08	0.65	0.84
noise	0	59.84	100.00 [100.00, 100.00]	99.84	1.00	1.00
noise	1e-06	59.84	99.96 [99.95, 99.97]	99.20	1.00	1.00
noise	1e-05	59.84	99.95 [99.93, 99.97]	98.99	1.00	1.00
noise	0.0001	59.84	99.90 [99.84, 99.94]	98.93	1.00	1.00
noise	0.001	60.00	99.54 [99.20, 99.73]	97.41	1.00	1.00
noise	0.01	59.69	96.58 [95.68, 97.35]	91.30	0.98	0.99
noise	0.05	58.75	83.94 [82.10, 85.70]	75.38	0.89	0.95
noise	0.1	52.97	65.47 [63.01, 67.82]	57.83	0.72	0.88
noise	0.2	31.41	36.35 [34.03, 38.76]	33.61	0.39	0.73
noise	0.3	15.16	21.76 [19.87, 23.59]	22.36	0.19	0.62
noise	0.4	6.72	15.85 [14.30, 17.44]	17.43	0.08	0.59
noise	0.5	3.12	13.06 [11.59, 14.55]	16.83	0.02	0.61
rotation	0	59.84	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	1e-06	59.84	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	1e-05	59.84	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	0.0001	59.84	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	0.001	59.84	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	0.01	59.84	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	0.05	59.69	94.27 [93.03, 95.34]	90.81	0.96	0.98
rotation	0.1	58.91	83.49 [81.46, 85.32]	78.45	0.88	0.94
rotation	0.2	56.72	68.69 [66.40, 70.98]	59.81	0.75	0.89
rotation	0.3	52.66	50.30 [48.01, 52.46]	44.89	0.58	0.81
rotation	0.4	51.88	52.19 [49.85, 54.53]	47.17	0.59	0.82
rotation	0.5	44.84	39.54 [37.43, 41.54]	36.62	0.45	0.75

Table 13
Full drift results for ResNet-18 on Tiny ImageNet (Grad-CAM). SSIM shown with bootstrap 95% CI.

Drift	γ	Acc.(%)	SSIM(%) [95% CI]	IoU(%)	Pearson	Cosine
blur	0	40.16	100.00 [100.00, 100.00]	99.84	1.00	1.00
blur	1e-06	16.56	33.11 [31.03, 35.25]	30.15	0.33	0.69
blur	1e-05	16.56	33.11 [31.03, 35.25]	30.15	0.33	0.69
blur	0.0001	16.56	33.11 [31.03, 35.25]	30.15	0.33	0.69
blur	0.001	16.56	33.11 [31.03, 35.25]	30.15	0.33	0.69
blur	0.01	16.56	33.11 [31.03, 35.25]	30.15	0.33	0.69
blur	0.05	16.56	33.11 [31.03, 35.25]	30.15	0.33	0.69
blur	0.1	16.56	33.11 [31.03, 35.25]	30.15	0.33	0.69
blur	0.2	9.22	26.35 [24.46, 28.24]	23.96	0.23	0.65

Table 13
Continued

Drift	γ	Acc.(%)	SSIM(%) [95% CI]	IoU(%)	Pearson	Cosine
blur	0.3	7.19	24.25 [22.56, 25.97]	22.00	0.19	0.63
blur	0.4	6.88	22.53 [20.95, 24.27]	20.38	0.16	0.62
blur	0.5	6.88	22.53 [20.95, 24.27]	20.38	0.16	0.62
brightness	0	40.16	100.00 [100.00, 100.00]	99.84	1.00	1.00
brightness	1e-06	40.16	99.98 [99.97, 99.99]	99.48	1.00	1.00
brightness	1e-05	40.16	99.80 [99.47, 99.97]	99.23	1.00	1.00
brightness	0.0001	40.16	99.87 [99.67, 99.98]	99.26	1.00	1.00
brightness	0.001	40.16	99.67 [99.26, 99.93]	98.52	1.00	1.00
brightness	0.01	40.31	98.51 [97.84, 99.07]	94.65	0.99	0.99
brightness	0.05	40.00	92.81 [91.39, 94.22]	85.42	0.94	0.97
brightness	0.1	39.84	87.40 [85.72, 89.03]	78.77	0.91	0.96
brightness	0.2	38.75	76.92 [74.92, 78.92]	68.09	0.83	0.92
brightness	0.3	36.56	67.00 [64.62, 69.33]	58.87	0.73	0.88
brightness	0.4	34.38	59.45 [57.15, 61.90]	52.10	0.65	0.84
brightness	0.5	33.12	53.90 [51.59, 56.18]	46.62	0.59	0.81
noise	0	40.16	100.00 [100.00, 100.00]	99.84	1.00	1.00
noise	1e-06	40.16	99.98 [99.97, 99.99]	99.51	1.00	1.00
noise	1e-05	40.16	99.88 [99.69, 99.98]	99.33	1.00	1.00
noise	0.0001	40.16	99.96 [99.94, 99.97]	99.27	1.00	1.00
noise	0.001	40.16	99.77 [99.71, 99.82]	98.09	1.00	1.00
noise	0.01	40.00	96.08 [94.97, 97.05]	90.87	0.97	0.99
noise	0.05	40.31	81.27 [79.33, 83.18]	72.00	0.87	0.94
noise	0.1	37.66	59.92 [57.49, 62.21]	52.15	0.66	0.84
noise	0.2	19.22	33.96 [31.81, 36.30]	31.68	0.35	0.72
noise	0.3	8.44	23.13 [21.48, 24.82]	22.82	0.19	0.66
noise	0.4	2.03	18.17 [16.84, 19.63]	19.62	0.12	0.64
noise	0.5	0.94	18.89 [17.51, 20.19]	19.85	0.11	0.65
rotation	0	40.16	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	1e-06	40.16	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	1e-05	40.16	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	0.0001	40.16	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	0.001	40.16	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	0.01	40.16	100.00 [100.00, 100.00]	99.84	1.00	1.00
rotation	0.05	40.62	86.04 [84.25, 87.69]	80.24	0.91	0.95
rotation	0.1	40.47	73.28 [70.97, 75.60]	66.42	0.79	0.90
rotation	0.2	40.62	83.29 [81.13, 85.32]	79.93	0.86	0.94
rotation	0.3	39.22	53.30 [50.85, 55.64]	46.57	0.59	0.82
rotation	0.4	35.78	41.82 [39.67, 44.12]	36.04	0.45	0.74
rotation	0.5	32.34	41.67 [39.39, 44.05]	37.45	0.46	0.76

Table 14
Full drift results for ShuffleNet v2 on Tiny ImageNet (Grad-CAM). SSIM shown with bootstrap 95% CI.

Drift	γ	Acc.(%)	SSIM(%) [95% CI]	IoU(%)	Pearson	Cosine
blur	0	53.75	100.00 [100.00, 100.00]	95.62	0.96	0.96
blur	1e-06	28.28	30.85 [28.67, 32.96]	20.66	0.23	0.50
blur	1e-05	28.28	30.85 [28.67, 32.96]	20.66	0.23	0.50
blur	0.0001	28.28	30.85 [28.67, 32.96]	20.66	0.23	0.50
blur	0.001	28.28	30.85 [28.67, 32.96]	20.66	0.23	0.50
blur	0.01	28.28	30.85 [28.67, 32.96]	20.66	0.23	0.50
blur	0.05	28.28	30.85 [28.67, 32.96]	20.66	0.23	0.50
blur	0.1	28.28	30.85 [28.67, 32.96]	20.66	0.23	0.50
blur	0.2	15.47	25.86 [23.88, 27.80]	16.92	0.15	0.46
blur	0.3	10.62	22.39 [20.52, 24.15]	14.14	0.10	0.43
blur	0.4	7.50	21.84 [20.07, 23.56]	13.81	0.09	0.43
blur	0.5	7.50	21.84 [20.07, 23.56]	13.81	0.09	0.43
brightness	0	53.75	100.00 [100.00, 100.00]	95.62	0.96	0.96
brightness	1e-06	53.75	99.87 [99.77, 99.95]	95.16	0.96	0.96
brightness	1e-05	53.75	99.81 [99.66, 99.91]	94.83	0.95	0.95
brightness	0.0001	53.75	99.76 [99.61, 99.87]	94.58	0.95	0.95
brightness	0.001	53.75	99.44 [99.22, 99.60]	93.27	0.95	0.95
brightness	0.01	53.12	95.54 [94.53, 96.45]	87.49	0.93	0.94
brightness	0.05	53.28	86.99 [85.44, 88.37]	75.66	0.87	0.90

Table 14
Continued

Drift	γ	Acc.(%)	SSIM(%) [95% CI]	IoU(%)	Pearson	Cosine
brightness	0.1	52.66	77.42 [75.47, 79.31]	64.39	0.79	0.85
brightness	0.2	49.06	65.56 [63.36, 67.71]	52.41	0.67	0.77
brightness	0.3	46.56	56.65 [54.48, 59.06]	42.38	0.56	0.70
brightness	0.4	41.72	48.67 [46.42, 50.91]	34.52	0.47	0.63
brightness	0.5	38.44	43.14 [40.92, 45.38]	29.12	0.39	0.59
noise	0	53.75	100.00 [100.00, 100.00]	95.62	0.96	0.96
noise	1e-06	53.75	99.85 [99.70, 99.95]	95.06	0.95	0.95
noise	1e-05	53.75	99.79 [99.64, 99.90]	94.73	0.95	0.95
noise	0.0001	53.75	99.60 [99.42, 99.75]	94.19	0.95	0.95
noise	0.001	53.75	98.14 [97.49, 98.67]	90.89	0.94	0.95
noise	0.01	53.91	89.59 [88.27, 90.93]	79.03	0.89	0.91
noise	0.05	52.19	63.09 [60.90, 65.21]	48.29	0.64	0.75
noise	0.1	44.38	39.62 [37.33, 41.90]	25.66	0.33	0.56
noise	0.2	20.62	21.63 [19.80, 23.51]	11.87	0.07	0.40
noise	0.3	5.16	12.21 [10.72, 13.74]	10.48	-0.02	0.43
noise	0.4	1.72	11.23 [9.79, 12.65]	11.18	-0.03	0.47
noise	0.5	0.31	10.24 [9.06, 11.54]	9.99	-0.05	0.45
rotation	0	53.75	100.00 [100.00, 100.00]	95.62	0.96	0.96
rotation	1e-06	53.75	100.00 [100.00, 100.00]	95.62	0.96	0.96
rotation	1e-05	53.75	100.00 [100.00, 100.00]	95.62	0.96	0.96
rotation	0.0001	53.75	100.00 [100.00, 100.00]	95.62	0.96	0.96
rotation	0.001	53.75	100.00 [100.00, 100.00]	95.62	0.96	0.96
rotation	0.01	53.75	100.00 [100.00, 100.00]	95.62	0.96	0.96
rotation	0.05	53.91	77.28 [75.12, 79.51]	66.52	0.76	0.83
rotation	0.1	53.75	71.95 [69.63, 74.49]	62.14	0.71	0.80
rotation	0.2	53.59	63.41 [60.96, 65.74]	50.22	0.64	0.76
rotation	0.3	51.41	60.41 [57.76, 63.06]	48.89	0.58	0.73
rotation	0.4	43.59	36.29 [34.23, 38.31]	23.73	0.30	0.55
rotation	0.5	46.56	36.57 [34.63, 38.56]	24.38	0.31	0.56

Table 15
Grad-CAM vs Grad-CAM++ paired comparison at the strongest intensity per drift type: Wilcoxon signed-rank p on the per-image SSIM difference and paired Cohen's d

Dataset	Model	Drift	γ	Wilcoxon p	Cohen's d
CIFAR-10	DenseNet-121	blur	0.5	1.04e-25	0.44
CIFAR-10	DenseNet-121	brightness	0.5	3.45e-19	0.33
CIFAR-10	DenseNet-121	noise	0.5	3.64e-31	0.49
CIFAR-10	DenseNet-121	rotation	0.5	6.18e-19	0.36
CIFAR-10	ResNet-18	blur	0.5	4.53e-55	0.72
CIFAR-10	ResNet-18	brightness	0.5	3.53e-35	0.50
CIFAR-10	ResNet-18	noise	0.5	4.62e-46	0.62
CIFAR-10	ResNet-18	rotation	0.5	1.95e-34	0.51
CIFAR-10	ShuffleNet v2	blur	0.5	1.58e-53	0.70
CIFAR-10	ShuffleNet v2	brightness	0.5	3.03e-35	0.49
CIFAR-10	ShuffleNet v2	noise	0.5	3.48e-56	0.74
CIFAR-10	ShuffleNet v2	rotation	0.5	2.22e-25	0.42
Tiny ImageNet	DenseNet-121	blur	0.5	6.27e-84	1.16
Tiny ImageNet	DenseNet-121	brightness	0.5	3.30e-36	0.55
Tiny ImageNet	DenseNet-121	noise	0.5	3.54e-88	1.21
Tiny ImageNet	DenseNet-121	rotation	0.5	5.88e-67	0.90
Tiny ImageNet	ResNet-18	blur	0.5	1.44e-97	1.50
Tiny ImageNet	ResNet-18	brightness	0.5	5.05e-101	1.27
Tiny ImageNet	ResNet-18	noise	0.5	3.43e-93	1.27
Tiny ImageNet	ResNet-18	rotation	0.5	1.35e-94	1.25
Tiny ImageNet	ShuffleNet v2	blur	0.5	3.91e-76	1.03
Tiny ImageNet	ShuffleNet v2	brightness	0.5	2.49e-94	1.36
Tiny ImageNet	ShuffleNet v2	noise	0.5	3.73e-90	1.25
Tiny ImageNet	ShuffleNet v2	rotation	0.5	4.36e-73	0.96

Table 16
IoU threshold-sensitivity at the strongest intensity per drift type: overlap recomputed at thresholds 0.3, 0.5 and 0.7

Dataset	Model	Drift	IoU@0.3	IoU@0.5	IoU@0.7
CIFAR-10	DenseNet-121	blur	51.54	37.83	28.84
CIFAR-10	DenseNet-121	brightness	77.86	70.69	65.10
CIFAR-10	DenseNet-121	noise	49.77	34.67	25.84
CIFAR-10	DenseNet-121	rotation	69.45	60.12	53.34
CIFAR-10	ResNet-18	blur	56.16	48.98	46.61
CIFAR-10	ResNet-18	brightness	77.84	75.37	74.66
CIFAR-10	ResNet-18	noise	57.25	52.00	50.61
CIFAR-10	ResNet-18	rotation	74.04	70.24	69.30
CIFAR-10	ShuffleNet v2	blur	45.88	34.48	27.45
CIFAR-10	ShuffleNet v2	brightness	64.58	57.08	51.88
CIFAR-10	ShuffleNet v2	noise	32.98	24.54	19.00
CIFAR-10	ShuffleNet v2	rotation	66.11	58.97	53.74
Tiny ImageNet	DenseNet-121	blur	35.29	22.72	13.87
Tiny ImageNet	DenseNet-121	brightness	61.33	51.08	41.57
Tiny ImageNet	DenseNet-121	noise	33.33	16.83	8.44
Tiny ImageNet	DenseNet-121	rotation	49.30	36.62	24.94
Tiny ImageNet	ResNet-18	blur	34.32	20.38	12.08
Tiny ImageNet	ResNet-18	brightness	55.83	46.62	39.84
Tiny ImageNet	ResNet-18	noise	36.43	19.85	9.98
Tiny ImageNet	ResNet-18	rotation	48.41	37.45	27.81
Tiny ImageNet	ShuffleNet v2	blur	21.58	13.81	9.33
Tiny ImageNet	ShuffleNet v2	brightness	36.25	29.12	24.18
Tiny ImageNet	ShuffleNet v2	noise	20.62	9.99	5.08
Tiny ImageNet	ShuffleNet v2	rotation	32.64	24.38	18.29

Acknowledgement
This research was funded by national funds through the FCT—Fundação para a Ciência e a Tecnologia, I.P., grants UIDB/04152/2020—Centro de Investigação em Gestão de Informação (MagIC) and UIDB/00315/2020—BRU-ISCTE-IUL.

Conflicts of Interest
The authors declare no conflicts of interest.

References
[1] Gama, Joao, Indre Zliobaite, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. (2014). A Survey on Concept Drift Adaptation. ACM Computing Surveys. https://doi.org/10.1145/2523813
[2] Widmer, Gerhard, and Miroslav Kubat. (1996). Learning in the Presence of Concept Drift and Hidden Contexts. Machine Learning. https://doi.org/10.1007/bf00116900
[3] Lu, Jie, Anjin Liu, Fan Dong, Feng Gu, Joao Gama, and Guangquan Zhang. (2018). Learning under Concept Drift: A Review. IEEE Transactions on Knowledge and Data Engineering. https://doi.org/10.1109/tkde.2018.2876857
[4] Eliwa, Entesar Hamed I., and Tarek Abd El-Hafeez. (2025). Deep Learning for Sustainable Agriculture: Automating Rice and Paddy Ripeness Classification for Enhanced Food Security. Egyptian Informatics Journal. https://doi.org/10.1016/j.eij.2025.100785
[5] Garcia-Rubio, Aiden M., Logan Heck, Kristin M. Schweitzer, Raymond M. Bateman, Adel Alaeddini, and Krystel K. Castillo-Villar. (2026). Detecting Concept Drift in Object Detection Models: A Collaborative AI-Human Approach for Defense Applications. IEEE Access. https://doi.org/10.1109/access.2026.3652106
[6] Liu, Chenruo, Kenan Tang, Yao Qin, and Qi Lei. (2025). Bridging Distribution Shift and AI Safety: Conceptual and Methodological Synergies. arXiv preprint arXiv:2505.22829. https://arxiv.org/abs/2505.22829.
[7] Brzezinski, Dariusz, and Jerzy Stefanowski. (2014). Combining Block-Based and Online Methods in Learning Ensembles from Concept Drifting Data Streams. Information Sciences. https://doi.org/10.1016/j.ins.2013.12.011
[8] Selvaraju, Ramprasaath R., Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. (2017). Grad-CAM: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision (pp. 618–626). https://doi.org/10.1109/iccv.2017.74
[9] Chattopadhay, Aditya, Anirban Sarkar, Prantik Howlader, and Vineeth N. Balasubramanian. (2018). Grad-CAM++:

Generalized gradient-based visual explanations for deep convolutional networks. In 2018 IEEE Winter Conference on Applications of Computer Vision (WACV) (pp. 839–847). IEEE. https://doi.org/10.1109/wacv.2018.00097

[10] Omeiza, Daniel, Skyler Speakman, Celia Cintas, and Komminist Weldermariam. (2019). Smooth Grad-CAM++: An Enhanced Inference Level Visualization Technique for Deep Convolutional Neural Network Models. arXiv preprint arXiv:1908.01224. https://arxiv.org/abs/1908.01224.

[11] Hinder, Fabian, Valerie Vaquet, Johannes Brinkrolf, and Barbara Hammer. (2023). Model-Based Explanations of Concept Drift. Neurocomputing. https://doi.org/10.1016/j.neucom.2023.126640

[12] Adebayo, Julius, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. (2018). Sanity Checks for Saliency Maps. Advances in Neural Information Processing Systems 31.

[13] Ross, Gordon J., Dimitris K. Tasoulis, and Niall M. Adams. (2011). Nonparametric Monitoring of Data Streams for Changes in Location and Scale. Technometrics. https://doi.org/10.1198/tech.2011.10069

[14] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 770–778). https://doi.org/10.1109/cvpr.2016.90

[15] Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. (2017). Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4700–4708). https://doi.org/10.1109/cvpr.2017.243

[16] Ma, Ningning, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. (2018). ShuffleNet V2: Practical guidelines for efficient CNN architecture design. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 116–131). https://doi.org/10.1007/978-3-030-01264-9_8

[17] Hovakimyan, Gurgen, and Jorge Miguel Bravo. (2024). Evolving Strategies in Machine Learning: A Systematic Review of Concept Drift Detection. Information. https://doi.org/10.3390/info15120786

[18] Baena-García, Manuel, José del Campo-Ávila, Raúl Fidalgo, Albert Bifet, Ricard Gavalda, and Rafael Morales-Bueno. (2006). Early Drift Detection Method. *In Fourth International Workshop on Knowledge Discovery from Data Streams*.

[19] Bifet, Albert, and Ricard Gavalda. (2007). Learning from Time-Changing Data with Adaptive Windowing. In Proceedings of the 2007 SIAM International Conference on Data Mining. https://doi.org/10.1137/1.9781611972771.42

[20] Raab, Christoph, Moritz Heusinger, and Frank-Michael Schleif. (2020). Reactive Soft Prototype Computing for Concept Drift Streams. Neurocomputing. https://doi.org/10.1016/j.neucom.2019.11.111

[21] Greco, Salvatore, Bartolomeo Vacchetti, Daniele Apiletti, and Tania Cerquitelli. (2025). Unsupervised Concept Drift Detection from Deep Learning Representations in Real-Time. IEEE Transactions on Knowledge and Data Engineering. https://doi.org/10.1109/tkde.2025.3593123

[22] Wiles, Olivia, Sven Gowal, Florian Stimberg, Sylvestre Alvise-Rebuffi, Ira Ktena, Krishnamurthy Dvijotham, and Taylan Cemgil. (2021). A Fine-Grained Analysis on Distribution Shift. arXiv preprint arXiv:2110.11328. https://arxiv.org/abs/2110.11328.

[23] Geirhos, Robert, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. (2018). ImageNet-Trained CNNs Are Biased towards Texture; Increasing Shape Bias Improves Accuracy and Robustness. In International Conference on Learning Representations.

[24] Liu, Zihe, Jie Lu, Junyu Xuan, and Guangquan Zhang. (2025). Learning Latent and Changing Dynamics in Real Non-Stationary Environments. IEEE Transactions on Knowledge and Data Engineering. https://doi.org/10.1109/tkde.2025.3535961

[25] Abdullahi, Mujaheed, Hitham Alhussian, Norshakirah Aziz, Said Jadid Abdulkadir, Yahia Baashar, Abdussalam Ahmed Alashhab, and Afroza Afrin. (2025). A Systematic Literature Review of Concept Drift Mitigation in Time-Series Applications. IEEE Access. https://doi.org/10.1109/access.2025.3587231

[26] Dong, Shi, Ping Wang, and Khushnood Abbas. (2021). A Survey on Deep Learning and Its Applications. Computer Science Review. https://doi.org/10.1016/j.cosrev.2021.100379

[27] O'Shea, Keiron, and Ryan Nash. (2015). An Introduction to Convolutional Neural Networks. arXiv preprint arXiv:1511.08458. https://arxiv.org/abs/1511.08458.

[28] Smilkov, Daniel, Nikhil Thorat, Been Kim, Fernanda Viegas, and Martin Wattenberg. (2017). SmoothGrad: Removing Noise by Adding Noise. arXiv preprint arXiv:1706.03825. https://arxiv.org/abs/1706.03825.

[29] Marmolejo-Saucedo, Jose Antonio, and Utku Kose. (2024). Numerical Grad-CAM Based Explainable Convolutional Neural Network for Brain Tumor Diagnosis. Mobile Networks and Applications. https://doi.org/10.1007/s11036-022-02021-6

[30] Teixeira, Brígida, Tiago Pinto, and Zita Vale. (2025). Detecting Concept Drift with SHapley Additive ExPlanations for Intelligent Model Retraining in Energy Generation Forecasting. In World Conference on Explainable Artificial Intelligence. https://doi.org/10.1007/978-3-032-08324-1_7

[31] Beemelmanns, Till, Shayan Sharifi, Manas Mehrotra, Ayushman Choudhuri, and Lutz Eckstein. (2026). Towards Trustworthy and Explainable AI for Perception Models: From Concept to Prototype Vehicle Deployment. arXiv preprint

arXiv:2605.16087. https://arxiv.org/abs/2605.16087.

[32] Jang, Moonjong, Seung-Ho Han, Ho-Jin Choi, and Byoung-Woong An. (2025). Improving Load Forecasting with Feature Selection via XAI and Expert-Guided Prompting. IEEE Access. https://doi.org/10.1109/access.2025.3637813

[33] Michaelis, Claudio, Benjamin Mitzkus, Robert Geirhos, Evgenia Rusak, Oliver Bringmann, Alexander S. Ecker, Matthias Bethge, and Wieland Brendel. (2019). Benchmarking Robustness in Object Detection: Autonomous Driving When Winter Is Coming. arXiv preprint arXiv:1907.07484. https://arxiv.org/abs/1907.07484.

[34] Recht, Benjamin, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. (2019). Do ImageNet classifiers generalize to ImageNet? In Proceedings of the International Conference on Machine Learning (pp. 5389–5400). PMLR. https://arxiv.org/abs/1902.10811.

[35] Krizhevsky, A., and Hinton, G. (2009). Learning multiple layers of features from tiny images. University of Toronto. kriz/cifar.html.

[36] Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. Advances in Neural Information Processing Systems 32.

[37] Duchi, John, Elad Hazan, and Yoram Singer. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*.

[38] Ruder, Sebastian. (2016). An Overview of Gradient Descent Optimization Algorithms. arXiv preprint arXiv:1609.04747. https://arxiv.org/abs/1609.04747.

[39] Harshit, N., and K. Mounvik. (2025). Improving Real-Time Concept Drift Detection Using a Hybrid Transformer-Autoencoder Framework. arXiv preprint arXiv:2508.07085. https://doi.org/10.36227/techrxiv.175393691.16483231/v1

[40] Piaseczny, Adam, Md Kamran Chowdhury Shisher, Shiqiang Wang, and Christopher Brinton. (2026). RCCDA: Adaptive Model Updates in the Presence of Concept Drift under a Constrained Resource Budget. Advances in Neural Information Processing Systems. https://doi.org/10.52202/085713-5183