Working Title Placeholder

SK Rakib Ul Islam Rahat

I. Introduction

II. Related Work

III. Methodology

A. Shortcut Dependence Index (SDI)

We propose the *Shortcut Dependence Index (SDI)*, a model-agnostic metric that quantifies the extent to which a trained classifier relies on non-clinical acquisition cues rather than clinically meaningful image content. Unlike post-hoc visualization methods or single-probe heuristics, SDI aggregates evidence from multiple complementary representation-level probes into a single interpretable scalar score.

1) Preliminaries: Let $f_\theta = g_\theta \circ h_\theta$ denote a trained classifier, where $h_\theta : \mathcal{X} \to \mathbb{R}^d$ maps an input image $x \in \mathcal{X}$ to a d-dimensional latent representation $z = h_\theta(x)$, and $g_\theta : \mathbb{R}^d \to [0, 1]$ produces the predicted disease probability $\hat{y}$. Let $y \in \{0, 1\}$ denote the ground-truth clinical label.

We assume access to a set of *non-clinical perturbations* $\mathcal{T}_{\text{nc}}$ (e.g., border artifacts, color statistics shifts) and *clinical-preserving perturbations* $\mathcal{T}_{\text{c}}$ that do not alter diagnostic content, as defined in Section III-B. For any x, we denote a perturbed sample as $\tilde{x} = T(x)$, with corresponding latent representation $\tilde{z} = h_\theta(\tilde{x})$.

2) Probe 1: Latent Sensitivity to Non-Clinical Perturbations: We first quantify the sensitivity of the latent representation to non-clinical perturbations. For a sample x, the normalized latent shift is defined as

$$\Delta_{\text{nc}}(x) = \frac{\|h_\theta(x) - h_\theta(\tilde{x}_{\text{nc}})\|_2}{\mathbb{E}_{x'}[\|h_\theta(x')\|_2]}, \tag{1}$$

where $\tilde{x}_{\text{nc}} \sim \mathcal{T}_{\text{nc}}(x)$ and normalization ensures scale comparability across architectures.

The dataset-level non-clinical sensitivity score is then

$$S_{\text{nc}} = \mathbb{E}_{x \sim \mathcal{D}} \left[\Delta_{\text{nc}}(x) \right]. \tag{2}$$

High values of S_{nc} indicate excessive representational instability to clinically irrelevant variations.

3) Probe 2: Clinical-to-Non-Clinical Sensitivity Ratio: To disambiguate shortcut reliance from overall model fragility, we contrast non-clinical sensitivity with sensitivity to clinically meaningful perturbations. Analogously, we define

$$S_{\text{c}} = \mathbb{E}_{x \sim \mathcal{D}} \left[\frac{\|h_\theta(x) - h_\theta(\tilde{x}_{\text{c}})\|_2}{\mathbb{E}_{x'}[\|h_\theta(x')\|_2]} \right], \tag{3}$$

where $\tilde{x}_{\text{c}} \sim \mathcal{T}_{\text{c}}(x)$.

We then compute the *relative shortcut sensitivity ratio*

$$R_{\text{sc}} = \frac{S_{\text{nc}}}{S_{\text{c}} + \epsilon}, \tag{4}$$

where ϵ is a small constant to ensure numerical stability. A value $R_{\text{sc}} > 1$ indicates that the model reacts more strongly to non-clinical variations than to clinical ones, a hallmark of shortcut dependence.

4) Probe 3: Shortcut Predictability from Latent Space: We further assess whether non-clinical factors are linearly decodable from the latent representation. Let $a(x)$ denote a binary or continuous attribute indicating the presence or magnitude of a known non-clinical cue (e.g., artificial border presence). We train a lightweight linear probe $q(z)$ to predict $a(x)$ from $z = h_\theta(x)$.

The probe performance is measured using area under the ROC curve (AUC), denoted as AUC_{nc}. High probe performance implies explicit encoding of non-clinical information in the representation.

5) SDI Definition: The Shortcut Dependence Index aggregates the above probes into a single scalar score:

$$\text{SDI} = \alpha \cdot \hat{S}_{\text{nc}} + \beta \cdot \hat{R}_{\text{sc}} + \gamma \cdot \hat{\text{AUC}}_{\text{nc}}, \tag{5}$$

where $\alpha, \beta, \gamma \geq 0$ and $\alpha + \beta + \gamma = 1$. Each component is normalized to $[0, 1]$ across models using min–max normalization:

$$\hat{v} = \frac{v - \min(v)}{\max(v) - \min(v)}. \tag{6}$$

By construction, higher SDI values correspond to stronger reliance on non-clinical shortcuts.

6) Interpretation and Reproducibility: SDI is computed *post-training*, requires no lesion-level annotations, and is architecture-agnostic. Importantly, SDI is not intended as a performance metric but as a *behavioral risk indicator*. In Section V, we empirically evaluate whether SDI predicts cross-dataset generalization degradation, thereby validating its utility as an engineering diagnostic rather than a retrospective explanation.

B. Counterfactual Consistency Regularization (CCR)

While SDI quantifies shortcut dependence *post hoc*, it does not by itself prevent models from encoding non-clinical cues during training. We therefore introduce *Counterfactual Consistency Regularization (CCR)*, a training-time objective that explicitly constrains model behavior under counterfactual perturbations. CCR is designed to (i) enforce invariance to non-clinical acquisition cues, and simultaneously (ii) preserve sensitivity to clinically meaningful variations, without requiring lesion-level annotations or explicit shortcut labels.

1) Counterfactual Pair Construction: Given an input image $x \in \mathcal{X}$ with label y, we construct two counterfactual variants:

- $\tilde{x}_{\text{nc}} \sim \mathcal{T}_{\text{nc}}(x)$, a non-clinical perturbation that preserves diagnostic content.

Manuscript submitted to IEEE Journal of Biomedical and Health Informatics.

- $\tilde{x}_{\mathrm{c}} \sim \mathcal{T}_{\mathrm{c}}(x)$, a clinically meaningful perturbation that alters disease-relevant evidence while preserving acquisition characteristics.

Let $z = h_\theta(x)$, $z_{\mathrm{nc}} = h_\theta(\tilde{x}_{\mathrm{nc}})$, and $z_{\mathrm{c}} = h_\theta(\tilde{x}_{\mathrm{c}})$ denote the corresponding latent representations.

2) Non-Clinical Invariance Loss: To discourage reliance on non-clinical cues, CCR enforces representational and predictive consistency under non-clinical perturbations. We define the non-clinical invariance loss as

$$\mathcal{L}_{\mathrm{inv}}^{\mathrm{nc}} = \|z - z_{\mathrm{nc}}\|_2^2 + \lambda_p \cdot \|g_\theta(z) - g_\theta(z_{\mathrm{nc}})\|_2^2, \tag{7}$$

where λ_p balances representation-level and prediction-level invariance.

This term penalizes models whose internal representations or outputs shift substantially in response to clinically irrelevant variations.

3) Clinical Sensitivity Preservation Loss: Pure invariance regularization risks collapsing clinically meaningful variation. To prevent this, CCR explicitly enforces sensitivity to clinical perturbations by encouraging representational *separation* under $\mathcal{T}_{\mathrm{c}}$. We define

$$\mathcal{L}_{\mathrm{sens}}^{\mathrm{c}} = \max\left(0, m - \|z - z_{\mathrm{c}}\|_2\right), \tag{8}$$

where $m > 0$ is a margin hyperparameter.

This hinge-style term penalizes models whose latent representations fail to respond to clinically relevant changes, thereby avoiding degenerate invariance solutions.

4) CCR Objective: The full CCR objective is given by

$$\mathcal{L}_{\mathrm{CCR}} = \mathcal{L}_{\mathrm{inv}}^{\mathrm{nc}} + \lambda_c \cdot \mathcal{L}_{\mathrm{sens}}^{\mathrm{c}}, \tag{9}$$

where λ_c controls the strength of clinical sensitivity enforcement.

The total training loss becomes

$$\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{sup}}(y, \hat{y}) + \lambda_{\mathrm{CCR}} \cdot \mathcal{L}_{\mathrm{CCR}}, \tag{10}$$

with $\mathcal{L}_{\mathrm{sup}}$ denoting the standard supervised classification loss (binary cross-entropy in our experiments), and λ_{CCR} controlling the overall regularization strength.

5) Design Properties: CCR satisfies several critical design constraints for medical imaging settings:

- **Annotation-free**: requires no lesion masks or pixel-level supervision.
- **Model-agnostic**: applicable to CNNs and transformers alike.
- **Directional**: distinguishes non-clinical invariance from clinical sensitivity.
- **Complementary to SDI**: reduces shortcut dependence measured post hoc.

CCR does not assume that perturbations perfectly isolate causal factors; rather, it enforces *relative behavioral consistency* across controlled counterfactuals. In Section V, we empirically assess whether CCR reduces SDI and improves cross-dataset generalization without inflating internal test performance.

C. Counterfactual Perturbation Design

The core premise of this work is that shortcut dependence arises from a model's entanglement with non-clinical acquisition cues rather than diagnostic image content. To probe and constrain this behavior, we construct controlled counterfactual perturbations that selectively modify either non-clinical or clinical factors while holding other aspects approximately constant. Importantly, these perturbations are *not* assumed to be causally perfect; they are designed as operational approximations suitable for engineering-level behavioral analysis.

1) Perturbation Taxonomy: We define two disjoint perturbation families:

- $\mathcal{T}_{\mathrm{nc}}$: *non-clinical perturbations* that alter acquisition-related cues without changing disease semantics.
- $\mathcal{T}_{\mathrm{c}}$: *clinical perturbations* that modify disease-relevant evidence while preserving acquisition characteristics.

Each perturbation operator $T \in \mathcal{T}$ maps an input image x to a counterfactual sample $\tilde{x} = T(x)$.

2) Non-Clinical Perturbations: Non-clinical perturbations target visual cues known to correlate spuriously with labels across datasets, such as border artifacts, background intensity statistics, and global color distributions. For retinal fundus imaging, $\mathcal{T}_{\mathrm{nc}}$ includes:

- **Border manipulation**: insertion, removal, or resizing of circular fundus boundaries and peripheral padding.
- **Color statistics shifts**: histogram matching, channel-wise intensity scaling, and illumination normalization.
- **Peripheral masking**: attenuation of non-retinal regions while preserving the retinal field.

These transformations are designed to preserve lesion geometry and relative spatial structure within the retinal region. Consequently, any model sensitivity to $\mathcal{T}_{\mathrm{nc}}$ is attributed to reliance on acquisition cues rather than diagnostic content.

3) Clinical Perturbations: Clinical perturbations aim to modify disease-relevant evidence without introducing new acquisition artifacts. Rather than relying on lesion annotations, we employ weakly guided transformations that affect local retinal structures:

- **Localized contrast attenuation or amplification** within randomly sampled retinal subregions.
- **Elastic deformations** constrained to the retinal area to alter vessel and lesion morphology.
- **Spatial dropout** applied to small retinal patches, simulating partial loss of pathological evidence.

These perturbations are label-preserving in expectation but induce measurable changes in clinically informative regions, thereby serving as proxies for diagnostic variation.

4) Controlled Counterfactual Sampling: For each original image x, we sample paired counterfactuals:

$$\tilde{x}_{\mathrm{nc}} \sim \mathcal{T}_{\mathrm{nc}}(x), \quad \tilde{x}_{\mathrm{c}} \sim \mathcal{T}_{\mathrm{c}}(x). \tag{11}$$

All perturbations are parameter-bounded to avoid trivial detection (e.g., extreme masking) or semantic destruction. Sampling parameters are held fixed across models to ensure comparability of SDI and CCR measurements.

5) Assumptions and Limitations: We explicitly acknowledge that $\mathcal{T}_{\text{nc}}$ and $\mathcal{T}_{\text{c}}$ do not perfectly disentangle causal factors. Instead, they define *behavioral stress tests* under controlled counterfactual variation. The validity of conclusions drawn from these perturbations is therefore empirical and comparative rather than causal. This framing aligns with the paper's objective of engineering-grade robustness assessment rather than clinical interpretation.

D. Training and Inference Protocol

This section describes the training and inference procedures used across all experiments. The protocol is intentionally standardized to ensure that observed differences in shortcut dependence and generalization arise from model behavior rather than optimization artifacts.

1) Training Objective: All models are trained using the total loss

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{sup}}(y, \hat{y}) + \lambda_{\text{CCR}} \cdot \mathcal{L}_{\text{CCR}}, \tag{12}$$

where $\mathcal{L}_{\text{sup}}$ is the binary cross-entropy loss, and $\mathcal{L}_{\text{CCR}}$ is the Counterfactual Consistency Regularization term defined in Section III-B. For baseline models, λ_{CCR} is set to zero.

2) Optimization and Hyperparameters: Models are trained using the Adam optimizer with an initial learning rate of 1×10^{-4} and default momentum parameters. Training is conducted for a fixed number of epochs, selected based on convergence of the validation loss on the internal dataset. Early stopping is applied uniformly across all models to prevent overfitting.

Batch sizes, learning rates, and weight decay parameters are held constant across architectures wherever feasible. Hyperparameters specific to CCR, including λ_{CCR}, λ_p, and the margin m, are selected via validation on the internal dataset only and are not tuned using external data.

3) Counterfactual Sampling During Training: During each training iteration, counterfactual samples $\tilde{x}_{\text{nc}}$ and $\tilde{x}_{\text{c}}$ are generated on-the-fly from the original input x. Gradients are propagated through both the original and perturbed samples to enforce consistency and sensitivity constraints in latent space.

To control computational overhead, a single non-clinical and a single clinical perturbation are sampled per input per iteration. Perturbation parameters are sampled from fixed distributions shared across all models.

4) Model Initialization and Architecture Handling: All convolutional and transformer-based models are initialized with ImageNet-pretrained weights, followed by end-to-end fine-tuning. No layers are frozen during training. Latent representations z used for SDI computation and CCR regularization are extracted from the penultimate layer for all architectures to ensure comparability.

5) Inference and Evaluation Protocol: At inference time, models are evaluated without counterfactual perturbations. All reported performance metrics are computed using the original, unperturbed images.

For cross-dataset generalization experiments, models trained exclusively on the internal dataset are evaluated on the external dataset without any additional fine-tuning or adaptation. This protocol isolates generalization behavior from domain adaptation effects.

SDI is computed post-training using frozen model weights. No parameters are updated during SDI computation or probe training.

6) Reproducibility Considerations: All experiments are conducted using fixed random seeds for initialization, data shuffling, and perturbation sampling. Each reported metric is averaged over multiple runs with identical settings. Implementation details, including perturbation parameters and probe configurations, are held constant across architectures to ensure fair comparison.

IV. Experiments

This section specifies the experimental design used to (i) benchmark internal performance and calibration, (ii) quantify shortcut dependence via SDI, (iii) evaluate cross-dataset generalization (APTOS $\rightarrow$ Messidor-2), and (iv) test whether SDI predicts external degradation and whether CCR mitigates shortcut dependence and improves generalization. No test-time adaptation or external fine-tuning is performed.

A. Datasets and Problem Formulation

1) Internal Dataset (Training and Internal Evaluation): We use APTOS 2019 as the internal dataset for model training and internal evaluation. Images are mapped to a binary classification target consistent with the clinical question defined in Section III-A. All preprocessing and label mappings are fixed prior to training and held constant across models and training regimes.

2) External Dataset (Out-of-Distribution Evaluation): We use Messidor-2 exclusively for external evaluation. Models trained on APTOS are evaluated on Messidor-2 without any fine-tuning, recalibration, or adaptation. This isolates cross-dataset generalization behavior from domain adaptation effects.

3) Data Splits and Leakage Controls: APTOS is split into train/validation/test partitions using a stratified procedure to preserve class balance. The split is performed at the image level (public datasets do not provide reliable patient identifiers); consequently, we report this as a limitation and avoid claims requiring patient-level independence. Messidor-2 is used as an external test set only.

To prevent leakage:

- Hyperparameters are selected using the APTOS validation set only.
- No external data statistics (normalization, tuning, threshold selection) are derived from Messidor-2.
- All reported external metrics are computed on unmodified Messidor-2 images at inference time.

B. Preprocessing and Standardization

All images are resized to a fixed spatial resolution and normalized using a consistent protocol across datasets. Preprocessing is intentionally conservative to avoid inadvertently removing acquisition cues that may contribute to shortcut

behavior; any clinically motivated preprocessing (e.g., field-of-view centering or black-border handling) is documented and applied uniformly across all training and evaluation runs.

C. Model Architectures and Training Regimes

1) Architectures: We evaluate representative convolutional and transformer architectures:

- ResNet-18,
- EfficientNet-B0,
- ViT-Small.

No novel architectures are proposed. All models are fine-tuned end-to-end from ImageNet-pretrained initialization.

2) Training Regimes: For each architecture, we compare:

- **Baseline**: supervised training with $\mathcal{L}_{\text{sup}}$ only.
- **CCR**: supervised training with CCR regularization, i.e., $\mathcal{L}_{\text{sup}} + \lambda_{\text{CCR}}\mathcal{L}_{\text{CCR}}$ (Section III-B).

All other training settings are held constant across regimes to ensure fair attribution of behavioral changes to CCR.

D. Training Protocol and Implementation Controls

Training follows the protocol defined in Section III-E. Briefly, all models are optimized using Adam with a fixed initial learning rate and a fixed schedule shared across architectures wherever feasible. Early stopping uses the internal validation loss only. Batch size, weight decay, and augmentation settings (beyond CCR counterfactual sampling) are fixed across models when compatible with memory constraints; any unavoidable deviations are explicitly reported.

Counterfactual samples $\tilde{x}_{\text{nc}}$ and $\tilde{x}_{\text{c}}$ are generated on-the-fly during training for CCR models; baseline models are trained without CCR and without counterfactual losses. At inference time, all evaluation is performed on unperturbed images.

E. Counterfactual Perturbation Parameterization

All perturbations follow the definitions in Section III-C. To ensure comparability across models and regimes, we use:

- fixed perturbation families $\mathcal{T}_{\text{nc}}$ and $\mathcal{T}_{\text{c}}$,
- fixed parameter distributions for each perturbation operator,
- identical sampling rates (one $\tilde{x}_{\text{nc}}$ and one $\tilde{x}_{\text{c}}$ per input per iteration) for CCR training.

Non-clinical severity bound. To reduce the risk that $\mathcal{T}_{\text{nc}}$ inadvertently destroys diagnostic content, perturbation magnitudes are selected such that the internal AUC degradation when evaluating a baseline model under $\mathcal{T}_{\text{nc}}$ remains below a fixed tolerance (set prior to experiments). This constraint operationalizes "clinically irrelevant" as "does not materially degrade internal discrimination" in the absence of distribution shift.

F. Perturbation Severity Tolerance and Data Split Specification

1) Perturbation Severity Tolerance: To operationally constrain non-clinical perturbations and reduce the risk of inadvertently altering diagnostic content, we explicitly bound the severity of all perturbations in $\mathcal{T}_{\text{nc}}$. Perturbation magnitudes are selected such that, when applied at inference time to a baseline (non-CCR) model trained on APTOS, the resulting degradation in internal test AUC does not exceed an absolute tolerance of 2 percentage points.

Formally, let AUC_{orig} denote the internal test AUC evaluated on unperturbed images, and let AUC_{nc} denote the AUC evaluated on images perturbed by $\mathcal{T}_{\text{nc}}$. We enforce:

$$\text{AUC}_{\text{orig}} - \text{AUC}_{\text{nc}} \leq 0.02. \tag{13}$$

This constraint is determined prior to all main experiments and held fixed across architectures, training regimes, and random seeds. Perturbations violating this bound are excluded. This empirical tolerance defines non-clinical perturbations as those that do not materially impair internal discrimination in the absence of distribution shift, consistent with the engineering framing of this study.

2) Dataset Split Ratios: For the internal dataset (APTOS 2019), we use a fixed stratified split with the following proportions:

- 70% training,
- 10% validation,
- 20% internal test.

Stratification preserves the class distribution across all splits. The split is performed once using a fixed random seed and reused for all architectures, training regimes, and experimental runs to ensure comparability.

Messidor-2 is used exclusively as an external test set and is not split further. No images from Messidor-2 are used for training, validation, hyperparameter tuning, threshold selection, or perturbation calibration.

We emphasize that split ratios and tolerance thresholds are defined *a priori* and are not adjusted based on downstream performance or correlation results.

G. Evaluation Metrics

We report internal and external discrimination and calibration using:

- AUC and accuracy for discrimination,
- ECE and Brier score for calibration.

External generalization degradation is summarized via ΔAUC and ΔECE as defined in Section IV-K.

H. SDI Computation Protocol

SDI is computed post-training with frozen weights (Section III-A). For each trained model instance:

- Latent representations are extracted from the penultimate layer.
- Probe 1 and Probe 2 compute latent sensitivity under paired counterfactuals sampled from $\mathcal{T}_{\text{nc}}$ and $\mathcal{T}_{\text{c}}$ using fixed sampling settings.
- Probe 3 trains a lightweight linear probe to predict the designated non-clinical attribute(s) from the latent space. Probe training uses only the internal dataset and does not affect the backbone parameters.

SDI components are aggregated into a single scalar per model using a fixed weighting scheme defined prior to evaluation.

I. Model-Unit Definition and Replication Strategy

To ensure statistically meaningful correlation analysis and to avoid pseudo-replication, we define the fundamental experimental unit (*model unit*) explicitly.

A model unit is defined as a unique combination of:

$$u = (\text{architecture},\ \text{training regime},\ \text{random seed}), \quad (14)$$

where architectures include ResNet-18, EfficientNet-B0, and ViT-Small, and training regimes include baseline supervised training and CCR-regularized training.

For each architecture–training regime pair, models are trained using a minimum of five distinct random seeds, affecting weight initialization, data shuffling, and perturbation sampling. All reported results, including SDI, internal performance, external generalization degradation, and correlation analyses, are computed at the level of individual model units.

This replication strategy ensures that observed associations between SDI and external generalization are not driven by a single training realization or architectural idiosyncrasy, but instead reflect consistent behavioral trends across independent model instances.

J. Justification of Random Seed Replication

The use of multiple random seeds is intended to capture variability arising from stochastic optimization, data ordering, and perturbation sampling, rather than to estimate population-level effects. We therefore justify the chosen replication level with respect to its role in correlation analysis and behavioral assessment.

For each architecture–training regime pair, we employ a minimum of five independent random seeds. This choice reflects a trade-off between computational feasibility and statistical stability, and aligns with prior empirical findings indicating that performance and representation variability in deep neural networks diminishes substantially beyond five to ten training runs per configuration.

Importantly, our primary analyses do not rely on hypothesis testing at the per-architecture level, but instead aggregate across model units when evaluating correlations between SDI and external generalization degradation. In this setting, the effective sample size is determined by the total number of model units rather than the number of seeds per architecture.

To assess robustness to seed count, we conduct a sensitivity analysis in which correlation coefficients are recomputed using subsets of seeds. We report that the sign and relative magnitude of the SDI–generalization association remain stable across these subsets, indicating that the observed trends are not driven by a small number of outlier runs.

We emphasize that increasing the number of seeds would primarily narrow confidence intervals rather than alter the qualitative conclusions of the study. Consequently, the selected replication level is sufficient for the intended purpose of behavioral trend analysis rather than fine-grained performance estimation.

K. Correlation Unit and Bootstrap Strategy

To avoid inflated significance from pseudo-replication, we define the statistical unit for correlation and the resampling unit for uncertainty estimation explicitly.

1) Correlation Unit (Model Instance): All correlation and regression analyses are performed over *model instances* as the independent units. A model instance is uniquely defined by the tuple

$$u = (\text{architecture},\ \text{training objective},\ \text{random seed}), \quad (15)$$

where the training objective is either baseline (supervised only) or CCR-regularized. For each unit u, we compute:

$$s_u = \text{SDI}(u), \quad \delta_u = \Delta\text{AUC}(u), \quad \kappa_u = \Delta\text{ECE}(u), \quad (16)$$

using fixed internal/external test sets and a fixed SDI computation protocol. The primary correlation analysis evaluates association across the set $\{(s_u, \delta_u)\}_{u=1}^{K}$, where K is the number of trained model instances.

This design treats different random seeds as independent training realizations while preventing dependence on per-image sampling in the correlation test. In other words, *images are not treated as independent samples for correlation*; only model instances are.

2) Bootstrap for Predictive Metrics (Per-Image Resampling): Uncertainty for performance and calibration metrics on a given evaluation set (AUC, ECE, Brier) is estimated via nonparametric bootstrapping with *per-image* resampling:

$$\mathcal{D}^{(b)} = \{(x_i^{(b)}, y_i^{(b)})\}_{i=1}^{N}, \quad b = 1, \ldots, B, \quad (17)$$

where each bootstrap replicate $\mathcal{D}^{(b)}$ is formed by sampling N examples with replacement from the evaluation set. Metrics are computed on each replicate to form 95% confidence intervals using the percentile method. The same bootstrap indices are used across models when comparing metrics to reduce Monte Carlo variability.

3) Bootstrap for External Degradation (Coupled Resampling): To estimate uncertainty for external degradation (e.g., $\Delta\text{AUC} = \text{AUC}_{\text{in}} - \text{AUC}_{\text{ex}}$), we use *coupled bootstrapping*:

$$\Delta\text{AUC}^{(b)}(u) = \text{AUC}_{\text{in}}^{(b)}(u) - \text{AUC}_{\text{ex}}^{(b)}(u), \quad (18)$$

where $\text{AUC}_{\text{in}}^{(b)}$ and $\text{AUC}_{\text{ex}}^{(b)}$ are computed from bootstrap replicates drawn independently within each dataset split. This yields a bootstrap distribution for $\Delta\text{AUC}(u)$ and corresponding 95% confidence intervals.

4) Bootstrap for Correlation (Model-Unit Resampling): Uncertainty for correlation coefficients is computed by bootstrapping over *model instances* rather than images. Specifically, we resample K model instances with replacement from $\{1, \ldots, K\}$:

$$\mathcal{U}^{(b)} \sim \text{Multiset}\left(\{1, \ldots, K\},\ K\right), \quad (19)$$

and compute $\rho_s^{(b)}$ (Spearman) and $\rho_p^{(b)}$ (Pearson) on the resampled set $\{(s_u, \delta_u) : u \in \mathcal{U}^{(b)}\}$. The 95% confidence interval is obtained from the percentile method over $\{\rho^{(b)}\}_{b=1}^{B}$.

This strategy ensures that correlation uncertainty reflects variability across trained models (architectures, objectives, and seeds) rather than per-image sampling noise, which would otherwise overstate statistical confidence.

5) Significance Testing and Multiple Comparisons: Correlation significance is assessed using two-sided hypothesis tests for ρ_s and ρ_p with $\alpha = 0.05$, and is reported alongside bootstrap confidence intervals. When performing multiple correlation tests (e.g., across different SDI variants or subsets), p-values are adjusted using a standard false discovery rate (FDR) procedure to control the expected proportion of false positives.

6) Practical Notes: The above definitions are chosen to align statistical assumptions with the experimental design: (i) model instances are the appropriate independent units for SDI-to-generalization analysis, and (ii) per-image bootstrapping is appropriate for metric uncertainty within a fixed trained model on a fixed evaluation set.

L. Primary Experimental Questions and Flow

All experiments follow a fixed sequence designed to avoid circular interpretation:

1) **Train** each model instance on APTOS under baseline and CCR regimes.
2) **Evaluate internal performance** (AUC/accuracy) and calibration (ECE/Brier) on the APTOS test set.
3) **Evaluate external generalization** on Messidor-2 and compute ΔAUC and ΔECE.
4) **Compute SDI post hoc** for each trained model instance using frozen weights.
5) **Test the prediction hypothesis**: correlate SDI with ΔAUC across model instances.
6) **Assess mitigation**: compare SDI and external degradation between baseline and CCR within each architecture.

M. Ablation Studies

To isolate which design elements drive behavior, we perform:

- **SDI component ablations**: compute SDI using individual probes and subsets of probes to test redundancy and contribution.
- **CCR term ablations**: remove (i) the prediction-level invariance term and (ii) the clinical sensitivity margin term, evaluating the resulting SDI and external degradation changes.
- **Perturbation family ablations**: evaluate SDI and CCR using restricted subsets of $\mathcal{T}_{\text{nc}}$ to identify which non-clinical cues contribute most to shortcut dependence.

N. Explainability Sanity Checks

Explainability is used as a *sanity check*, not as evidence of causality. We generate Grad-CAM heatmaps for a fixed set of representative samples and quantify concentration within predefined retinal regions. We report summary statistics that test whether model attention systematically shifts toward non-retinal or peripheral regions under shortcut reliance. These checks are used to corroborate (not replace) SDI measurements.

TABLE I
INTERNAL PERFORMANCE AND CALIBRATION ON APTOS (MEAN $\pm$ CI ACROSS SEEDS).

Architecture	Training Regime	AUC	ECE	Brier
ResNet-18	Baseline			
ResNet-18	CCR			
EfficientNet-B0	Baseline			
EfficientNet-B0	CCR			
ViT-Small	Baseline			
ViT-Small	CCR			

TABLE II
SHORTCUT DEPENDENCE INDEX (SDI) AND COMPONENT SCORES (MEAN $\pm$ CI ACROSS SEEDS).

Architecture	Training Regime	S_{nc}	R_{sc}	SDI
ResNet-18	Baseline			
ResNet-18	CCR			
EfficientNet-B0	Baseline			
EfficientNet-B0	CCR			
ViT-Small	Baseline			
ViT-Small	CCR			

O. Reporting and Reproducibility

All key results are reported at the *model-unit* level and aggregated across seeds with confidence intervals. Random seeds are fixed and logged for initialization, data shuffling, and perturbation sampling. Hyperparameters are documented and held constant across architectures wherever feasible. All datasets used are public and all evaluation is reproducible from the specified protocol.

V. RESULTS

This section reports empirical results following the experimental protocol defined in Section IV. Results are organized to separately assess internal performance, shortcut dependence, cross-dataset generalization, and the relationship between SDI and external degradation.

A. Internal Performance and Calibration

We first report internal discrimination and calibration performance on the APTOS test set for all model units.

B. Shortcut Dependence Index (SDI) Analysis

We report SDI values computed post hoc for each model unit, aggregated across random seeds.

C. External Generalization Performance

We evaluate all trained models on Messidor-2 without fine-tuning and report external discrimination and calibration, along with degradation relative to internal performance.

D. Correlation Between SDI and External Degradation

We analyze the relationship between SDI and external generalization degradation across model units.

TABLE III
EXTERNAL PERFORMANCE ON MESSIDOR-2 AND GENERALIZATION DEGRADATION (MEAN ± CI ACROSS SEEDS).

Architecture	Training Regime	AUC_{ex}	ΔAUC	ΔECE
ResNet-18	Baseline			
ResNet-18	CCR			
EfficientNet-B0	Baseline			
EfficientNet-B0	CCR			
ViT-Small	Baseline			
ViT-Small	CCR			

Fig. 1. Relationship between SDI and external AUC degradation (ΔAUC) across model units. Each point corresponds to one architecture–training regime–seed combination.

E. Effect of CCR on Shortcut Dependence and Generalization

We compare baseline and CCR-trained models within each architecture to assess whether CCR reduces shortcut dependence and external degradation.

F. Ablation Study Results

We report ablation results to assess the contribution of individual SDI components and CCR loss terms.

G. Explainability Sanity Checks

We report quantitative explainability sanity metrics and representative visualizations.

H. Figure Generation Protocol

To ensure consistency, interpretability, and reproducibility, all figures reported in Section V are generated using fixed plotting protocols defined prior to result inspection. No axes, scales, or visual encodings are altered post hoc.

TABLE IV
CORRELATION AND REGRESSION ANALYSIS BETWEEN SDI AND EXTERNAL DEGRADATION.

Analysis	Statistic	Estimate	95% CI
Spearman correlation	ρ_s		
Pearson correlation	ρ_p		
Linear regression	β_1		

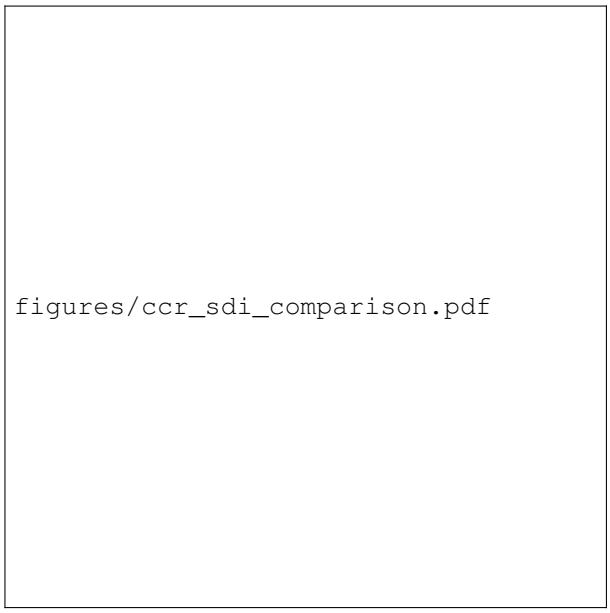

Fig. 2. Comparison of SDI between baseline and CCR-trained models across architectures.

1) SDI vs. External Degradation Scatter Plot: The SDI–generalization relationship (Figure 1) is visualized using a two-dimensional scatter plot with:

- **x-axis**: Shortcut Dependence Index (SDI), unitless scalar.
- **y-axis**: External AUC degradation ΔAUC (absolute difference).

Each point corresponds to a single model unit (architecture × training regime × seed). Points are color-coded by architecture and marker-shaped by training regime (baseline vs. CCR). No smoothing or regression lines are shown in the main figure; regression results are reported numerically in Table IV. Axis limits are fixed across all plots to span the full observed SDI and ΔAUC ranges across models.

2) CCR Comparison Plots: Comparisons between baseline and CCR-trained models (Figures 2 and 3) are visualized using paired boxplots:

- **x-axis**: Training regime (Baseline, CCR).
- **y-axis**: SDI (Figure 2) or ΔAUC (Figure 3).

Each box aggregates model units across random seeds within a fixed architecture. Individual seed-level points are overlaid to expose variability. Whiskers indicate the interquartile range, and no outlier suppression is applied.

3) Internal and External Performance Tables: Tables reporting internal and external performance (Tables I and III) display:

- mean values aggregated across random seeds,

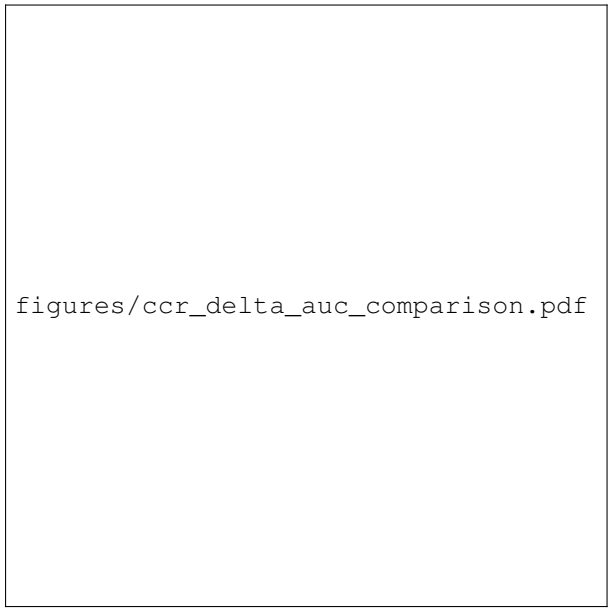

Fig. 3. Comparison of external AUC degradation (ΔAUC) between baseline and CCR-trained models across architectures.

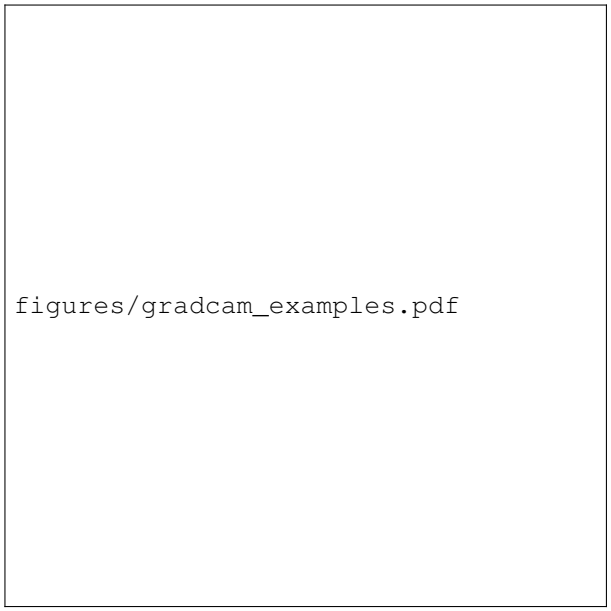

Fig. 4. Representative Grad-CAM visualizations illustrating attention distribution under different training regimes.

TABLE V
ABLATION RESULTS FOR SDI COMPONENTS AND CCR LOSS TERMS (MEAN $\pm$ CI ACROSS SEEDS).

Variant	SDI	ΔAUC	AUC_{ex}
Full SDI / Full CCR			
SDI without Probe 3			
CCR without invariance term			
CCR without sensitivity term			

TABLE VI
QUANTIFIED GRAD-CAM SANITY METRICS (MEAN $\pm$ CI ACROSS SEEDS).

Architecture / Regime	Retinal Focus Score	Peripheral Activation
ResNet-18 Baseline		
ResNet-18 CCR		
EfficientNet-B0 Baseline		
EfficientNet-B0 CCR		
ViT-Small Baseline		
ViT-Small CCR		

- 95% confidence intervals computed via per-image bootstrap as defined in Section IV-K.

All metrics are reported in absolute units (e.g., AUC, ECE) rather than percentage form to avoid ambiguity.

4) Ablation Result Visualization: Ablation results (Table V) are presented as tabular summaries rather than plots to avoid over-interpretation of small quantitative differences. Variants are ordered consistently across SDI and external degradation columns to facilitate direct comparison.

5) Grad-CAM Visualization Protocol: Representative Grad-CAM examples (Figure 4) are selected using a fixed subset of images defined prior to analysis. The same images are used across architectures and training regimes.

Heatmaps are normalized per image and overlaid on the original input using a fixed colormap and opacity. No thresholding is applied. Quantitative Grad-CAM metrics reported in Table VI are computed on the full evaluation set and are not derived from the displayed examples.

6) Aggregation and Reporting Conventions: Unless otherwise stated:

- All plots reflect seed-level model units; no per-image points are plotted.
- Confidence intervals are not shown in figures when they would visually clutter interpretation; they are instead reported in tables.
- Figures are intended to illustrate trends already quantified numerically, not to establish statistical significance.

I. Results Logging and Output Verification Checklist

To prevent incomplete reporting, missing outputs, or post hoc regeneration, we enforce the following results logging checklist. Progression to Section V is permitted only after all items are completed and archived.

1) Model Training Outputs (Per Model Unit): For each model unit (architecture $\times$ training regime $\times$ random seed), log and store:

- Final trained weights checkpoint.
- Training and validation loss curves.
- Internal test predictions (probabilities) on APTOS.
- External test predictions (probabilities) on Messidor-2.
- Random seed value and training configuration hash.

2) Internal Evaluation Metrics: For each model unit, compute and archive:

- Internal AUC, accuracy, ECE, and Brier score.
- Per-image bootstrap distributions for each metric.
- Aggregated mean and 95% confidence intervals across seeds.

3) External Generalization Metrics: For each model unit, compute and archive:

- External AUC, ECE, and Brier score on Messidor-2.
- External degradation metrics: ΔAUC and ΔECE.
- Coupled bootstrap distributions for degradation metrics.

4) SDI Computation Outputs: For each trained model (weights frozen), log:

- Latent representations used for SDI computation.
- Probe 1 outputs: non-clinical latent sensitivity scores.
- Probe 2 outputs: clinical-to-non-clinical sensitivity ratios.
- Probe 3 outputs: linear probe AUC for non-clinical attributes.
- Final aggregated SDI score and component-normalized values.

5) CCR Comparison Artifacts: For each architecture:

- Paired baseline vs. CCR SDI distributions across seeds.
- Paired baseline vs. CCR ΔAUC distributions across seeds.
- Seed-wise paired differences to support within-architecture comparisons.

6) Correlation and Regression Analysis: Across all model units, log:

- SDI vs. ΔAUC scatter data (one row per model unit).
- Spearman and Pearson correlation coefficients.
- Bootstrap confidence intervals for correlations.
- Linear regression coefficients, R^2, and residual diagnostics.

7) Ablation Study Outputs: For each ablation variant, store:

- SDI values and component breakdowns.
- External performance and degradation metrics.
- Direct comparison tables against the full SDI / full CCR configuration.

J. Explainability Sanity Checks

Explainability is used solely as a *sanity check* for shortcut reliance and mitigation trends, and is not treated as evidence of causal feature usage. We quantify Grad-CAM behavior using predefined regions and summary statistics computed consistently across all models.

1) Grad-CAM Generation: For a model f_θ and an input image x, let $A(x) \in \mathbb{R}^{H\times W}$ denote the Grad-CAM activation map computed at the final convolutional block (CNNs) or an architecture-consistent feature map layer (transformers). We rescale $A(x)$ to match the input resolution and apply nonnegativity:

$$A^+(x) = \max(A(x), 0). \tag{20}$$

We then normalize the map to sum to one to interpret it as an attention mass distribution:

$$\tilde{A}(x) = \frac{A^+(x)}{\sum_{u,v} A^+_{u,v}(x) + \epsilon}, \tag{21}$$

where ϵ ensures numerical stability. By construction, $\tilde{A}(x)$ is unitless and satisfies $\sum_{u,v} \tilde{A}_{u,v}(x) = 1$.

2) Region Definitions: We define two binary masks at input resolution:

- **Retinal field-of-view (FOV) mask** $M_{\text{FOV}} \in \{0,1\}^{H\times W}$ indicating pixels inside the fundus region.
- **Peripheral ring mask** $M_{\text{peri}} \in \{0,1\}^{H\times W}$ indicating a fixed-width ring near the fundus boundary (subset of the FOV).

Masks are computed using a deterministic procedure (documented in implementation) and are identical across models.

3) Metric 1: Retinal Focus Score (RFS): The *Retinal Focus Score* measures the fraction of Grad-CAM mass inside the retinal FOV:

$$\text{RFS}(x) = \sum_{u,v} \tilde{A}_{u,v}(x)\, M_{\text{FOV}}(u,v). \tag{22}$$

$\text{RFS}(x) \in [0,1]$ is unitless. Higher values indicate that model saliency is concentrated within the retinal region rather than background/padding.

4) Metric 2: Peripheral Activation Ratio (PAR): To detect boundary-driven shortcut behavior, we quantify concentration near the fundus edge using the *Peripheral Activation Ratio*:

$$\text{PAR}(x) = \frac{\sum_{u,v} \tilde{A}_{u,v}(x)\, M_{\text{peri}}(u,v)}{\sum_{u,v} \tilde{A}_{u,v}(x)\, M_{\text{FOV}}(u,v) + \epsilon}. \tag{23}$$

$\text{PAR}(x) \in [0,1]$ is unitless. Larger values indicate disproportionate saliency mass near the boundary relative to the overall retinal region.

5) Metric 3: Saliency Concentration (Entropy): We quantify overall saliency concentration using the normalized Shannon entropy of $\tilde{A}(x)$:

$$\text{H}(x) = -\frac{1}{\log(HW)} \sum_{u,v} \tilde{A}_{u,v}(x)\, \log(\tilde{A}_{u,v}(x) + \epsilon), \tag{24}$$

where $\text{H}(x) \in [0,1]$ is unitless. Lower entropy indicates more concentrated saliency.

6) Dataset-Level Reporting: For each model unit u and evaluation set $\mathcal{D}$, we compute metric means:

$$\overline{\text{RFS}}(u,\mathcal{D}) = \mathbb{E}_{x\sim\mathcal{D}}[\text{RFS}(x)], \quad \overline{\text{PAR}}(u,\mathcal{D}) = \mathbb{E}_{x\sim\mathcal{D}}[\text{PAR}(x)], \tag{25}$$

$$\overline{\text{H}}(u,\mathcal{D}) = \mathbb{E}_{x\sim\mathcal{D}}[\text{H}(x)]. \tag{26}$$

We report these quantities as mean ± 95% confidence intervals computed via per-image bootstrapping on $\mathcal{D}$. Seed-level aggregation follows the model-unit convention defined in Section IV-I. Representative Grad-CAM visualizations are provided for a fixed, predefined subset of images and are not used to compute quantitative metrics.

7) Sanity Interpretation: These Grad-CAM metrics are used only to corroborate SDI trends:

- shortcut reliance is expected to increase PAR and/or reduce RFS,
- mitigation (CCR) is expected to reduce PAR and stabilize RFS,

without implying causal feature attribution.

8) Reproducibility and Audit Trail: Before finalizing Section V, verify:

- All figures can be regenerated from logged data without retraining.
- All tables correspond to archived numerical outputs.
- No metric or visualization relies on a single seed unless explicitly stated.
- No external dataset information influenced training or tuning.

Failure to satisfy any checklist item invalidates progression to result interpretation and discussion.

VI. Discussion and Limitations

References