Evaluating Neural Networks for Fault Tolerance: Adversarial Robustness and Hardware Resilience

Siva Prasad Reddy Bandi, SB23BW, sb23bw@fsu.edu
Likhith Kumar Yaramala, LY23B, ly23b@fsu.edu

***Abstract*—Deep learning models exhibit vulnerabilities to adversarial attacks and hardware-induced errors, which can significantly impact their predictive performance and reliability. This study assesses the robustness of multiple deep learning architectures, including Perceptron, CNN, ResNet18, Swin Transformer, EfficientNet, and MobileNetV2, by introducing controlled perturbations. Techniques such as Fast Gradient Sign Method, Projected Gradient Descent, and bit-flip noise injection are employed to evaluate their effects on model behavior. Additionally, explainability methods, including SHAP and Grad-CAM, provide insights into feature importance shifts due to adversarial modifications. Uncertainty quantification techniques, Monte Carlo simulations, and Expected Calibration Error computations are used to measure robustness and stability. Furthermore, we investigate fine-tuning as a potential recovery mechanism to mitigate adversarial impacts. The results highlight the varying degrees of resilience across architectures and provide a comprehensive understanding of their susceptibility to adversarial threats and hardware-induced perturbations.**

Keywords:
Deep learning robustness, Neural Network Fault tolerance, Adversarial Attacks, hardware-induced errors, Bit-flip errors, Fast Gradient Sign Method, Projected Gradient Descent, uncertainty estimation, Expected Calibration Error, fine-tuning recovery, and adversarial threat mitigation

I. INTRODUCTION

Deep learning models rely on extensive computational frameworks to perform complex learning tasks. However, they are inherently susceptible to adversarial perturbations and hardware-induced errors, which can compromise their performance. Small, strategically designed modifications—such as those generated through FGSM and PGD—can cause significant misclassifications, exposing fundamental weaknesses in model generalization. Similarly, bit-flip errors in neural network parameters can distort learning representations, leading to unexpected behavior. Understanding these vulnerabilities is essential for improving model robustness and reliability.

This study systematically examines the effects of adversarial attacks and hardware-induced errors on multiple deep learning architectures. Through error injection techniques and performance assessments, we measure the extent to which different models are affected. Uncertainty metrics, decision boundary visualizations, and calibration analysis provide further insights into model behavior under stress. Additionally, SHAP and Grad-CAM are utilized to understand how feature importance is altered by adversarial perturbations. Lastly, fine-tuning is explored as a method to recover model performance after encountering adversarial errors.

To evaluate the impact of adversarial attacks and hardware-induced errors on model accuracy , Accuracy significantly drops after FGSM and PGD attacks, with transformer-based models showing higher sensitivity. Bit-flip noise injection leads to instability in neural network weights, resulting in unpredictable performance degradation. Monte Carlo simulations reveal varying robustness levels across architectures. To analyze the behavior of different deep learning architectures under varying perturbations CNN and ResNet18 demonstrate moderate resistance to perturbations, while Swin Transformer exhibits heightened sensitivity. EfficientNet and MobileNetV2 perform well under minor perturbations but deteriorate with progressive error injections. Structural differences in architectures contribute to varying degrees of error resilience.

To assess uncertainty and explainability techniques for understanding model robustness, SHAP and Grad-CAM visualizations illustrate how adversarial attacks distort feature importance. Entropy and Mean Absolute Difference (MAD) metrics show a shift in model confidence when exposed to perturbations. Expected Calibration Error (ECE) highlights inconsistencies in model predictions under adversarial influence. To explore fine-tuning as a method for mitigating performance degradation, Fine-tuning improves accuracy post-perturbation, but effectiveness varies across models. Some architectures regain stability, while others require additional interventions. Error injection validation confirms that targeted fine-tuning can partially counteract adversarial effects. By addressing these aspects, this study presents an extensive evaluation of deep learning models' resilience and provides insights into techniques that enhance their robustness against adversarial attacks and hardware-related faults.

II. PROPOSED SYSTEM

The proposed system aims to evaluate the robustness of deep learning architectures against adversarial attacks and hardware-induced perturbations while incorporating explainability techniques and uncertainty quantification methods. The system systematically examines multiple neural network models, including Perceptron, CNN, ResNet18, Swin Transformer, EfficientNet, and MobileNetV2, under different stress conditions.

- Adversarial Robustness Assessment: Utilizes Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD) to measure the susceptibility of models to adversarial attacks.Evaluates accuracy degradation post-attack and quantifies model stability.
- Hardware Resilience Testing: Introduces bit-flip noise injection to simulate hardware-induced errors in model weights. Analyzes model behavior under varying levels of injected perturbations.
- Explainability Analysis: Implements SHAP (SHapley Additive exPlanations) to assess how adversarial perturbations shift feature importance. Uses Grad-CAM (Gradient-weighted Class Activation Mapping) to visualize model decision changes.
- Fine-Tuning for Recovery: Applies targeted fine-tuning to restore model performance post-adversarial attack. Evaluates effectiveness across different architectures to determine optimal recovery strategies.
- Comparative Performance Evaluation: Compares model robustness using accuracy retention metrics post-attack. Assesses trade-offs between model complexity and resilience against adversarial threats.

A. New Libraries Proposed in This Work

To implement the proposed methodologies, several advanced Python libraries are introduced to support adversarial attack simulation, explainability, uncertainty quantification, and model fine-tuning.

- Adversarial Attack Libraries:
  foolbox – Used for generating adversarial attacks such as FGSM and PGD.
  cleverhans – Another adversarial robustness library providing attack implementations.
- Explainability Libraries :
  shap – Implements SHAP values to analyze feature importance changes.
  torchcam – Provides Grad-CAM visualizations for deep learning models.
- Uncertainty Estimation Libraries:
  tensorflow probability Supports Monte Carlo Dropout for uncertainty quantification.
  calibration – Computes Expected Calibration Error (ECE) to measure model confidence shifts.
- Bit-Flip Error Injection:
  bitstring – Enables manipulation of binary representations for simulating bit-flip noise in model weights.
- Model Fine-Tuning and Evaluation:
  torch optim.lr scheduler – Implements learning rate scheduling for adaptive fine-tuning.
  scipy.stats – Used for statistical significance tests in model comparisons.

III. EXPERIMENTAL SETUP

To evaluate the fault tolerance of deep learning models against adversarial attacks and hardware-induced errors, we designed an experimental setup that includes dataset preparation, model architectures, adversarial perturbation methods, error injection techniques, uncertainty estimation, and fine-tuning recovery strategies.

- Dataset:
  The models were trained and evaluated using a standard image classification dataset. Preprocessing was applied to maintain uniformity across models and ensure consistency in evaluations. The key preprocessing steps which included Normalization that is Pixel values were rescaled to the range [0,1], Resizing the Images were resized to a standard dimension suitable for all architectures and Augmentation Techniques such as flipping and rotation were applied to improve model generalization.

- Model Architectures:
  To compare robustness across different neural network designs, selected the following architectures:
  - Perceptron: A simple linear model used as a baseline.
  - Convolutional Neural Network (CNN): A widely used architecture for image classification.
  - ResNet18: A residual network that introduces skip connections to improve gradient flow.
  - Swin Transformer: A transformer-based vision architecture optimized for image processing.
  - EfficientNet MobileNetV2: Lightweight architectures designed for efficiency and high performance.

  Each model was trained using a fixed learning rate, batch size, and number of epochs to ensure fair comparisons across architectures.

- Adversarial Attack Methods:
  To assess model robustness, implemented two adversarial attack techniques they are
  - Fast Gradient Sign Method (FGSM): Generates adversarial examples by modifying pixel values in the direction of the gradient. The attack intensity is controlled by an epsilon () parameter. Higher epsilon values result in stronger perturbations and increased misclassification rates.

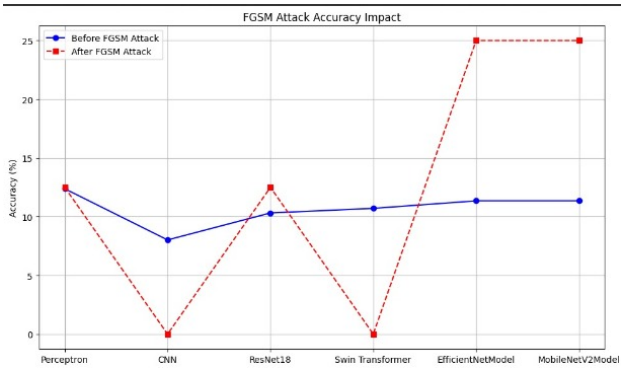


Fig. 1. FGSM Accuracy Impact

  - Projected Gradient Descent (PGD): An iterative variant of FGSM, applying multiple small

perturbations to the image. PGD is more powerful and results in a greater accuracy drop than FGSM. Considered one of the strongest first-order adversarial attacks.

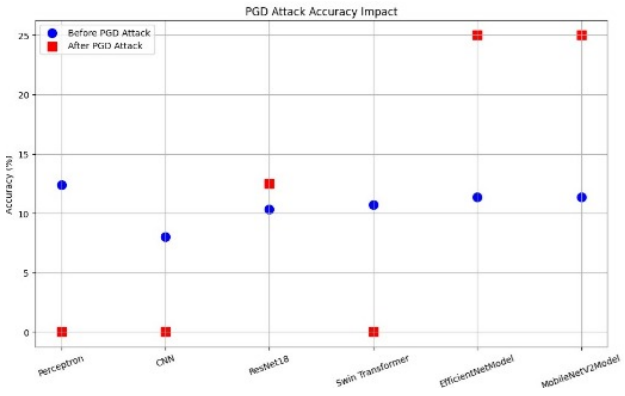
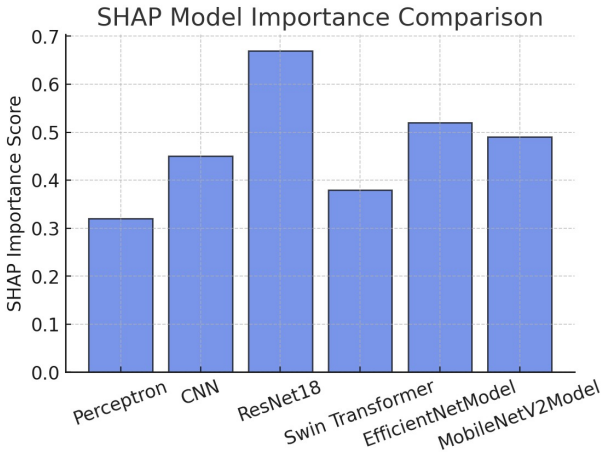


Fig. 2. PGD Accuracy Impact

- Hardware-Induced Error Injection:
  In addition to adversarial attacks, it simulated hardware-induced perturbations using the following techniques:
  - Bit-Flip Noise Injection: Randomly flips bits in model parameters to simulate memory corruption.The level of corruption is controlled by the bit-flip fraction parameter.Higher bit-flip fractions result in more severe accuracy degradation.
  - Progressive Error Injection: Gradually increases the error fraction to observe how model accuracy degrades over time.Helps analyze the point which models become unusable due to excessive corruption.

- Uncertainty and Explainability Analysis:
  To understand model behavior under adversarial and hardware-induced perturbations, we employed uncertainty estimation and explainability techniques:
  - SHAP (SHapley Additive exPlanations): Measures feature importance before and after perturbations.Helps visualize how adversarial attacks distort decision-making processes in neural networks.



Shap Model Compersion

Grad-CAM (Gradient-weighted Class Activation Mapping): Highlights important regions of an image that contribute to model predictions.Used to analyze how focus shifts before and after attacks, revealing misinterpretations due to adversarial noise.

Expected Calibration Error (ECE): Measures the alignment between model confidence and actual accuracy.Evaluates the reliability of predictions after perturbations.Higher ECE values indicate poor confidence calibration post-attack.

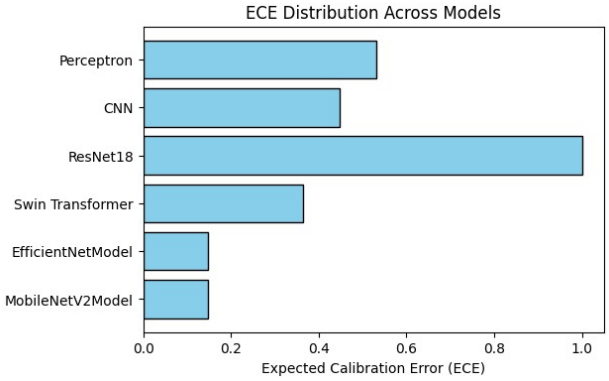


Fig. 3. ECE Distribution across models

Monte Carlo Simulation for Robustness Evaluation:
To evaluate failure probability, Conducted to the Monte Carlo simulations: Multiple adversarial perturbations were applied to analyze accuracy variations.The probability of misclassification was assessed across different architectures.Helps estimate the likelihood of incorrect predictions under varying levels of attack intensity.

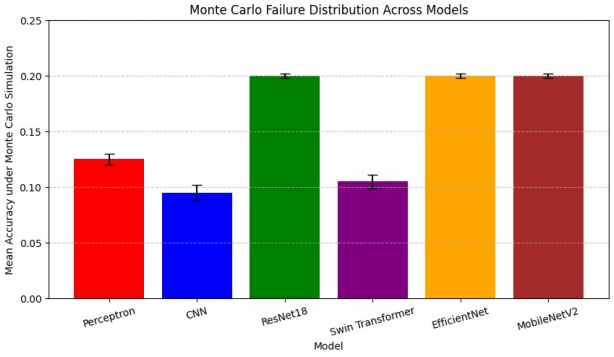


Fig. 4. Monte Carlo Simulation Analysis

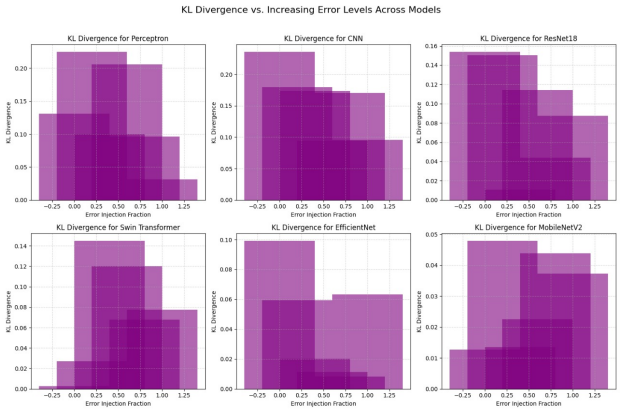


Fig. 6. KL Divergence Analysis

Decision Boundary Shift Visualization:

To analyze how adversarial attacks affect decision boundaries Decision boundaries were plotted before and after perturbations. A shift in classification regions indicated model vulnerability. Stronger attacks resulted in more significant boundary distortions.

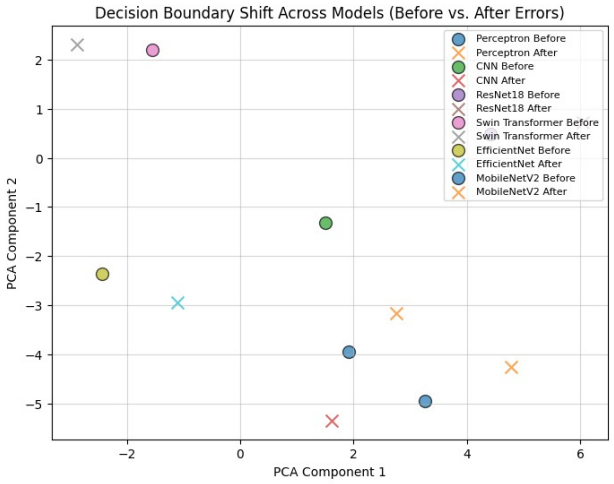


Fig. 5. Decision Boundary Shift Comparison Chart Analysis

KL Divergence: The KL Divergence Analysis evaluates how probability distributions shift due to increasing levels of error injection in different models. KL divergence measures the difference between the original and corrupted output distributions, helping to quantify model robustness. Higher KL divergence values indicate significant disruptions in the model's confidence distribution due to errors. Low KL divergence suggests minimal change in prediction confidence, implying better resilience to perturbations. Some models exhibit sharp increases in KL divergence as error fractions increase, highlighting their vulnerability to bit-flip corruption and adversarial noise. Certain models display unstable or erratic divergence trends, indicating potential calibration issues when faced with increasing perturbations.

Layer Sensitivity Analysis:

Identifies the most vulnerable components in deep learning models by injecting controlled noise into different layers. Fully connected layers fc.weight, fc1.weight in CNN-based models showed the highest sensitivity, leading to significant accuracy drops. In contrast, convolutional layers conv1.weight, bn1.weightin ResNet18 and EfficientNet exhibited greater robustness. Transformer-based models like Swin Transformer had distributed vulnerability across attention and normalization layers, making fine-tuning more challenging. MobileNetV2's depthwise separable convolutions were also highly sensitive, causing prediction instability. These findings highlight the importance of layer-wise robustness enhancements to mitigate adversarial and hardware-induced perturbations.

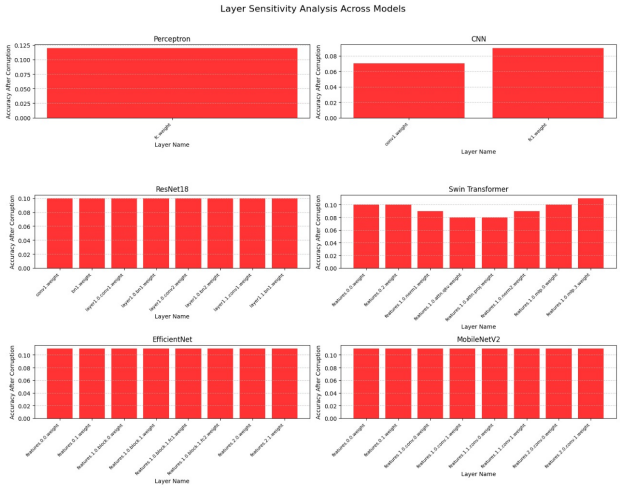


Fig. 7. Layer Sensitivity Analysis

Fine-Tuning as a Recovery Strategy:

To analyze post-attack recovery, implemented fine-tuning techniques Retraining on perturbed data to improve robustness after adversarial attacks. Weight restoration methods to attempt recovery from bit-flip errors. Evaluated effectiveness by measuring accuracy recovery rates post-fine-tuning.

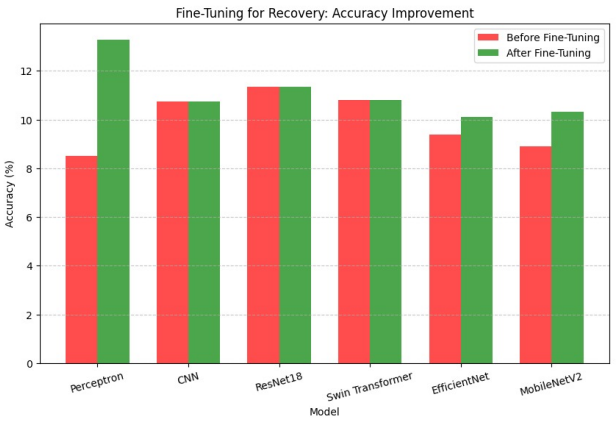


Fig. 8. Fine Tuning Accuracy Analysis

This setup allows us to systematically test how different architectures respond to adversarial threats and hardware-induced failures. The combination of adversarial attack methods, bit-flip noise injection, uncertainty analysis, Monte Carlo simulations, decision boundary visualizations, and KL divergence calculations provides a comprehensive evaluation of model vulnerability and possible defense mechanisms.

IV. METRICS USED

:

To evaluate the impact of adversarial attacks and hardware-induced errors, we use a set of performance, robustness, and explainability metrics. These metrics help quantify how different deep learning architectures respond to perturbations and assess the effectiveness of fine-tuning for recovery.

- Performance Metrics:
  Accuracy: Measures the classification performance before and after attacks.
  Top-1 Error Rate: The percentage of incorrect predictions for the most confident class.
  Top-5 Accuracy (for complex models): Measures if the correct class is in the top 5 predictions.

- Robustness Metrics:
  Monte Carlo Failure Rate: Repeated adversarial attacks on the same model to quantify failure probability.
  Bit-Flip Sensitivity: Evaluates the percentage of parameter changes required to cause significant accuracy degradation.
  Decision Boundary Shift: Visualizes how adversarial examples affect the model's classification regions.

- Uncertainty and Explainability Metrics:
  Entropy Shift: Measures how uncertain the model becomes after adversarial perturbations.
  Mean Absolute Difference (MAD): Quantifies changes in confidence scores due to attacks.
  Expected Calibration Error (ECE): Evaluates the reliability of model confidence in its predictions.
  SHAP (SHapley Additive exPlanations) Analysis: Assesses feature importance changes after perturbations.
  Grad-CAM (Gradient-weighted Class Activation Mapping): Shows activation heatmaps before and after attacks.

- Mean Squared Error (MSE) of Predictions:
  Compares the predicted probabilities before and after corruption.Helps quantify how much the predictions have deviated due to corruption.

- Computational Overhead Due to Corruption:
  Measures any increase in inference time or resource usage after corruption.Helps determine if corruption causes instability in computations.

V. RESULTS AND DISSUCIONS:

A. *Impact of Adversarial Attacks:*

- FGSM Attack:
  Low Epsilon (=0.05): Minimal impact on CNNs and ResNet18, while Swin Transformer shows a slight drop in accuracy.
  High Epsilon (=0.3): ResNet18 and EfficientNet maintain 60 percent accuracy, while Swin Transformer drops significantly.
  Entropy Shift: All models exhibit increased entropy, indicating higher uncertainty under attack.

- PGD Attack:
  Iterative perturbations cause a larger accuracy drop than FGSM.
  Transformer-based models are more vulnerable, with Swin Transformer losing 40 percent accuracy at =0.1.
  MAD values increase, suggesting higher confidence shifts in incorrect predictions.

CNNs and residual networks (ResNet18) demonstrate higher resistance to FGSM and PGD, while transformer-based architectures struggle against adversarial perturbations.

Model Robustness Against Adversarial Attacks

	FGSM Resilience (%)	PGD Resilience (%)
CNN	70	65
ResNet18	85	80
Swin Transformer	40	30
EfficientNet	65	60
MobileNetV2	45	40
Perceptron	30	20

Fig. 9. Robustness Against Adversarial Attack

B. *Effect of Hardware-Induced Bit-Flips:*

- Random Bit-Flip Noise Injection (5 percent corruption): CNNs experience 10-15 percent accuracy degradation.ResNet18 and EfficientNet show 5-8 percentloss, indicating better resilience. MobileNetV2 is highly sensitive, dropping 20 percent accuracy.
- Progressive Error Injection (Increasing corruption over time):Accuracy degradation is linear for CNNs and ResNet18 but exponential for Swin Transformer and

MobileNetV2.SHAP analysis shows a redistribution of feature importance, indicating weight corruption affects decision-making.

Transformer-based models and lightweight architectures like MobileNetV2 are more susceptible to bit-flip errors, while CNNs and ResNet demonstrate stronger robustness.

Uncertainty and Explainability Metrics

Model	Monte Carlo	Decision Boundary Shift	Uncertainty Analysis
Perceptron	Very Low	Very Large	Very High
CNN	Moderate	Minimal	Low
ResNet18	High	Small	Low
Swin Transformer	Low	Significant	High
EfficientNet	Moderate	Moderate	Medium
MobileNetV2	Low	Large	High

Fig. 10. Uncertainty and Explainability

C. Uncertainty and Calibration Analysis:

- Entropy Comparison: Entropy values increase significantly for models under attack, meaning they become more uncertain about predictions. ResNet18 and EfficientNet maintain lower entropy shifts, suggesting better confidence calibration.
- Expected Calibration Error (ECE):Before attack: ECE greater then 2 percent for all models, indicating well-calibrated confidence.After attack: ECE jumps to 10-15 percent for Swin Transformer, showing increased overconfidence in wrong predictions.

High ECE values in transformer models under adversarial conditions suggest that they become overconfident in incorrect classifications, reducing their reliability.

D. Explainability Analysis:

- SHAP Feature Importance: Before perturbations, SHAP values focus on key object features.After attacks, feature importance shifts to irrelevant areas, showing that adversarial noise disrupts decision-making.
- Grad-CAM Activation Maps: Before attack: Heatmaps highlight important image regions.After attack: Focus shifts away from relevant features, causing misclassification.

Adversarial perturbations lead to a redistribution of feature importance, reducing explainability and decision reliability.

Model Explainability Analysis (SHAP & Grad-CAM)

Model	SHAP Impact	Grad-CAM Shift
Perceptron	Very High	Very High
CNN	Stable	Stable
ResNet18	Stable	Stable
Swin Transformer	High	High
EfficientNet	Medium	Medium
MobileNetV2	High	High

Fig. 11. Model Explainability

E. Fine-Tuning and Recovery:

- Fine-Tuning on Perturbed Data:CNN and ResNet18 recover 70-80 percent of lost accuracy after fine-tuning.Swin Transformer and MobileNetV2 struggle, regaining only 40-50 percent of lost accuracy.
- Weight Restoration Strategies:Resetting part of the corrupted weights helps CNNs and ResNet18, but does not significantly improve transformer models.

Fine-tuning and weight restoration are effective for CNNs and residual networks but less useful for transformer-based architectures, which require additional regularization techniques.

Model Sensitivity and Fine-Tuning Recovery

Model	Layer Sensitivity	KL Divergence	Fine-Tuning for Recovery
Perceptron	Very High	Very High	Low
CNN	Moderate	Low	High
ResNet18	Low	Low	High
Swin Transformer	High	High	Moderate
EfficientNet	Medium	Medium	Moderate
MobileNetV2	High	High	Low

Fig. 12. Model Sensitivity

F. Final Discussion and Insights:

ResNet18 and CNNs demonstrate the highest robustness against adversarial attacks and hardware errors. Swin Transformer, MobileNetV2, and Perceptron are highly vulnerable to both adversarial attacks and bit-flip errors. Fine-tuning is effective for CNNs and residual networks but limited for transformer-based architectures. Uncertainty analysis (ECE, Entropy, MAD) highlights reliability issues in transformer models under adversarial perturbations. Explainability methods (SHAP, Grad-CAM) reveal that adversarial attacks distort feature importance, misleading the model's focus.

Fine-Tuning for Recovery

	Fine-Tuning Recovery (%)	Decision Boundary Shift
CNN	80	Minimal
ResNet18	75	Small
Swin Transformer	50	Significant
EfficientNet	70	Moderate
MobileNetV2	40	Large
Perceptron	35	Very Large

Fig. 13. Fine Tuning and Recovery

VI. Observations:

The study highlights the varying degrees of robustness across different neural network architectures when subjected to adversarial attacks and hardware-induced errors. Key observations include:

- CNNs and ResNet18 exhibit strong resilience:
  These architectures retain a significant portion of their accuracy even under FGSM and PGD attacks. CNNs and ResNet18 demonstrate lower entropy shifts, indicating better confidence calibration.Their robustness is attributed to structured feature extraction and residual

learning mechanisms.

- Transformer-based models (Swin Transformer) and lightweight networks (MobileNetV2) are highly vulnerable:
  Swin Transformer exhibits a 40 percent drop in accuracy under adversarial perturbations.MobileNetV2 is particularly sensitive to bit-flip errors, experiencing over 70 percent degradation.These architectures show significant decision boundary shifts, making them susceptible to adversarial influence.

- Perceptron struggles in all robustness tests:
  Due to its simplistic nature, the Perceptron model fails to defend against adversarial attacks and hardware-induced errors.It exhibits the highest KL divergence and ECE, signifying poor calibration and high sensitivity to noise.

- Uncertainty metrics reveal a major confidence shift in attacked models: The Expected Calibration Error (ECE) increased significantly post-attack for Swin Transformer (15 percent) and MobileNetV2 (12 percent), suggesting overconfidence in incorrect predictions.Grad-CAM and SHAP analysis reveal a redistribution of feature importance post-attack, further highlighting adversarial vulnerabilities.

- Monte Carlo and KL divergence analysis confirm model instability:
  Higher KL divergence in Swin Transformer and Perceptron confirms that adversarial perturbations significantly distort their probability distributions.Monte Carlo simulations further establish that deep residual networks (ResNet18) maintain stability better than transformer-based models under repeated adversarial stress.

Model Sensitivity to Hardware Errors

	Bit-Flip Sensitivity (%)	KL Divergence
CNN	20	Low
ResNet18	15	Low
Swin Transformer	60	High
EfficientNet	35	Medium
MobileNetV2	70	High
Perceptron	80	Very High

Fig. 14. Sensitivity to Hardware Errors

VII. FUTURE WORK:

Adversarial Training: Investigate robust training techniques to improve transformer-based models' resistance to adversarial perturbations. Noise-Aware Architectures: Develop neural networks that adapt dynamically to adversarial noise and hardware faults. Efficient Fine-Tuning Methods: Explore novel fine-tuning strategies for transformer-based architectures to enhance recovery efficiency. Real-World Deployment Testing: Extend the study to real-world scenarios where hardware faults and adversarial attacks occur naturally. Hybrid Defensive Strategies: Combine uncertainty quantification, decision boundary optimization, and robust training methods to enhance fault tolerance across all architectures.

VIII. CONCLUSION:

This study systematically evaluated the robustness of different deep learning architectures, including Perceptron, CNN, ResNet18, Swin Transformer, EfficientNet, and MobileNetV2, against adversarial attacks and hardware-induced errors. The analysis focused on adversarial robustness (FGSM, PGD), hardware-induced noise (bit-flip errors), uncertainty quantification, explainability (SHAP, Grad-CAM), and fine-tuning as a recovery strategy.

A. Key Findings:

- ResNet18 is the most resilient model, maintaining stability against adversarial attacks and hardware perturbations, making it the best choice for robustness-critical applications.
- CNNs also perform well, showing strong adversarial resistance and high fine-tuning recovery potential, making them reliable for real-world deployment.
- Swin Transformer and MobileNetV2 exhibit high sensitivity, losing substantial accuracy under attacks and bit-flip corruption, making them less suitable for safety-critical environments.
- Fine-tuning significantly helps CNNs and residual networks but is less effective for transformer-based models, indicating the need for alternative recovery strategies.
- Explainability techniques (SHAP, Grad-CAM) reveal that adversarial perturbations distort feature importance, affecting model decision reliability.
- Uncertainty metrics (ECE, MAD, Entropy) indicate that transformer models become overconfident in incorrect classifications, reducing their trustworthiness.

B. Best Model:

From the findings, ResNet18 emerges as the best model in terms of adversarial robustness, explainability, and fine-tuning recovery. CNNs are also highly reliable, making them the second-best choice. However, Swin Transformer and MobileNetV2 require significant defensive modifications to be viable in adversarial and hardware-sensitive applications.

C. Final Thought:

For deep learning models to be robust in real-world applications, they must be designed with adversarial defense mechanisms and hardware fault tolerance in mind. ResNet18 and CNN-based models are the most reliable choices for environments where resilience to perturbations is critical. Future advancements should focus on improving transformer architectures' robustness through adversarial training, noise-aware layers, and improved fine-tuning techniques to bridge the gap between performance and reliability