Revolutionary Advances in Computing and Electronics (RACE): An International Journal
Volume 02, Issue 01, pp.34-46, 2026
ISSN (Online): 3107-8540

Received: 22/01/2026, Accepted: 27/04/20026

Explainable Robustness Against Localized Malware Obfuscation: A Hybrid CNN-ViT Approach with Comparative Attention Analysis

Ayush Raj, Shadan Ahmad, Pranjal Kumar Jha

School of Computer Science Engineering and Technology, Bennett University, Greater Noida, India

e23cseu0277@bennett.edu.in, e23cseu0280@bennett.edu.in, e23cseu0287@bennett.edu.in

ABSTRACT

The rapid proliferation of obfuscated and zero-day malware variants poses a critical challenge to modern cybersecurity defences. Traditional Convolutional Neural Network (CNN) classifiers, while effective on clean binary images, exhibit catastrophic vulnerability when adversaries apply localised packing or byte-level scrambling to evade detection. In this paper, a hybrid deep learning architecture that combines a lightweight CNN feature extractor with a Vision Transformer (ViT) encoder to achieve explainable robustness against localised malware obfuscation. The CNN extracts hierarchical texture patterns from grayscale binary images, while the ViT models long-range spatial dependencies through multi-head self-attention across 196 image patches. Crucially, this approach injects Spatial Dropout (nn.Dropout2d) into the CNN layers and employs Focal Loss to handle severe class imbalance, forcing the model to learn from partially corrupted feature maps during training itself. Evaluated on the Malimg benchmark (9,339 samples, 25 families), the hybrid model achieves 93.80% accuracy with a Macro AUC of 0.9980. Under simulated obfuscation with 30% local byte noise, the hybrid architecture maintains 81.82% accuracy, while an equivalent pure CNN baseline using the same CNN feature extractor collapses to 52.62%, a gap of 29.20 percentage points. Through comparative attention heatmaps, this paper provides visual, interpretable proof that while CNN activations scatter across noise artifacts under obfuscation, the ViT's self-attention dynamically shifts to focus on persistent global structural payloads, explaining *why* the architecture survives where CNNs fail.

Keywords: *Malware Classification, Vision Transformer, Convolutional Neural Network, Explainable AI, Obfuscation Robustness, Focal Loss, Spatial Dropout*

1. Introduction

The global cybersecurity landscape faces an unprecedented escalation in malware threats, with over 450,000 new malicious programs detected daily [1]. To evade traditional signature-based detection, adversaries increasingly employ advanced obfuscation techniques such as packing, encryption, and oligomorphic code mutation, creating highly evasive "zero day" variants that bypass static analysis [2].

A seminal advancement in proactive detection is malware binary visualization [3], which converts executable binaries into 2D grayscale images by interpreting byte sequences as pixel intensities. This enables the direct application of deep learning-based classification without requiring disassembly or complex feature engineering. Recently, Convolutional Neural Networks (CNNs) have achieved strong accuracy on benign visual malware datasets [4, 5].

*Corresponding author: Ayush Raj, School of Computer Science Engineering and Technology, Bennett University, Greater Noida, India (e23cseu0277@bennett.edu.in)

However, a growing body of recent literature [6, 7] highlights a critical flaw: CNNs exhibit extreme vulnerability to localized obfuscation. Their strictly localized receptive fields cause them to overfit to local byte textures. When these textures are scrambled by packing or section level encryption, CNN feature activations degrade catastrophically, leading to classification failure.

Brosolo et al. [7] recently demonstrated that obfuscation techniques can reduce CNN based malware classification accuracy by up to 50%, and proposed explainability-driven mitigation strategies using tools such as Occlusion Maps, HiResCAM, and SHAP. Their work established that understanding *why* models fail under obfuscation is as important as improving raw accuracy. Concurrently, Asam et al. [8] introduced an explainable hybrid CNN-Transformer framework (ConvNeXt-Swin) that leveraged Grad- CAM to visualize learned representations, achieving 94.04% accuracy on Malimg. However, their work did not evaluate robustness under obfuscation conditions.

To address this gap, this study adopts a hybrid architecture integrating a lightweight CNN with a Vision Transformer (ViT) [9]. Unlike CNNs, ViTs leverage multi head self attention to capture global spatial dependencies across the entire sequence of image patches, providing intrinsic mathematical ro- bustness against localized noise perturbations. This work bridges the two open fronts identified in recent literature: (a) the need for obfuscation robust architectures, and (b) the demand for explainable evidence of *why* a model survives adversarial manipulation.

The contributions are as follows:

1. This study designs a robust, lightweight hybrid CNN-ViT (3.4M parameters) with **Spatial Dropout** (nn.Dropout2d) injected into the CNN feature extractor, pre conditioning the ViT to learn from imperfect local patches.

2. this study employs **Focal Loss** [10] with inverse frequency class weights to resolve extreme class imbalance, specifically fixing the previously zero performing *Autorun.K* class.

3. this study conducts a **controlled obfuscation experiment** using an equivalent CNN baseline (identical CNNFeatureExtractor architecture, 1.18M params) to isolate the ViT's contribution, proving a 29.20 pp accuracy advantage at 30% noise.

4. this study presents **Comparative Attention Analysis** under obfuscation, providing visual, inter- pretable proof that the ViT ignores noise signatures and locks onto persistent global payloads explaining the mechanism behind the robustness.

The remainder of this paper details the related work, proposed methodology, experiments, and the core explainability analysis.

2. Related Work

2.1. Malware Binary Visualization

The concept of visualising malware binaries as images was introduced by Nataraj et al. [3], who demonstrated that malware samples from the same family exhibit visually similar grayscale patterns when their byte sequences are rendered as pixel arrays. This foundational work led to the creation of the Malimg dataset, which has since become a standard benchmark in the field. The binary visualisation approach transforms the malware classification problem

into an image recognition task, enabling the direct application of computer vision techniques without requiring complex feature engineering or disassembly [11].

2.2. CNN-Based Malware Classification and Obfuscation Vulnerability

Since deep learning has shown success in vision problems, a number of studies have used CNNs on malware images. Kalash et al. [4] obtained 98.52% on Malimg with a VGG-16 network and Vasan et al. [5] suggested IMCFN, a collection of CNNs with 98.82%. However, Gibert et al. [6] emphasized that these scores are obtained on unobfuscated *clean* benchmarks. Malware developers in the real world continuously use polymorphic packing, section-level encryption and code metamorphism to scramble local byte patterns. [2].

Most importantly, Brosolo et al. [7] introduced the first empirical study of CNN tolerance to obfuscated data, which showed that accuracy decreased by as much as 50% and suggested explainability-based mitigation through HiResCAM and SHAP analysis. Their work showed that the black-boxiness of CNNs has a root cause in overreliance on local texture features that are trivially easy to destroy by scrambling at the binary representation level.

2.3. Hybrid CNN-Transformer Architectures

Vision Transformers (ViTs) [9] learn global interactions via self-attention over patch sequences and overcome the locality limitation of CNNs. CNN feature extraction combined with the Transformer-based sequence modelling has proven helpful: CoAtNet. [12] entangles convolution and attention for data-effective learning, whereas Khan et al. [13] introduced LeViT MC, a lightweight ViT for malware classification.

Asam et al. [8] proposed an explainable hybrid framework based on the ConvNeXt Swin Transformer, achieving 94.04% on Malimg with Grad-CAM-based interpretability, which is most applicable to this work. They did not, however, test under obfuscation conditions; only clean data was evaluated by them: this leaves the significant question unanswered: is the global attention of the Transformer, in fact, faithful to adversarial manipulation?

This question is directly answered in this work. This work not only demonstrates the robustness advantage empirically but also provides comparative attention heatmaps as visual proof of why the Transformer component survives where the CNN perishes.

3. Proposed Methodology

Four sequential stages comprise the hybrid CNN ViT architecture: (1) a CNN feature extractor with Spatial Dropout for local pattern recognition; (2) a patch embedding layer that transforms spatial feature maps into token sequences; (3) a Vision Transformer encoder for global context modeling; and (4) a classification head Fig. 1 depicts the overall architecture.

3.1. CNN Feature Extractor with Spatial Dropout

Each of the four convolutional blocks that make up the CNN component has two 3x 3 convolution layers with Batch Normalization and GELU activation, followed by 2x2 max pooling. The input resolution is
224× 224, and the channel progression is 1 → 32 → 64 → 128 → 256, resulting in feature maps with a spatial resolution of 14x14.

Critically, this approach injects **Spatial Dropout** (nn.Dropout2d, p = 0.2) after each activation function. Unlike standard dropout, which zeroes individual neurons, Spatial Dropout drops entire 2D feature maps, forcing the downstream ViT encoder to learn robust

representations from incomplete local information. This design prepares the model for obfuscation scenarios in which local texture patches are corrupted or missing.

3.2. Patch Embedding

The CNN output tensor of shape (B, 256, 14, 14) is re- formed into the series of 196 dimension 256 to- kens that are where the points of the feature map are associated with the individual tokens of the dataset. grid. The refinement of the normalization is done by a linear projection using Layer Normalization. patch representations:

$$z_i = \text{LayerNorm}(W_p \cdot f_i + b_p), \quad i = 1, \ldots, 196 \tag{1}$$

and f_i is a flattened feature at spatial position i.

3.3. Vision Transformer Encoder

A patch sequence is preceded by discoverable [CLS] trained as a model that is token and learned positionally:

$$Z_0 = [z_{cls}; z_1; \ldots; z_{196}] + E_{pos} \tag{2}$$

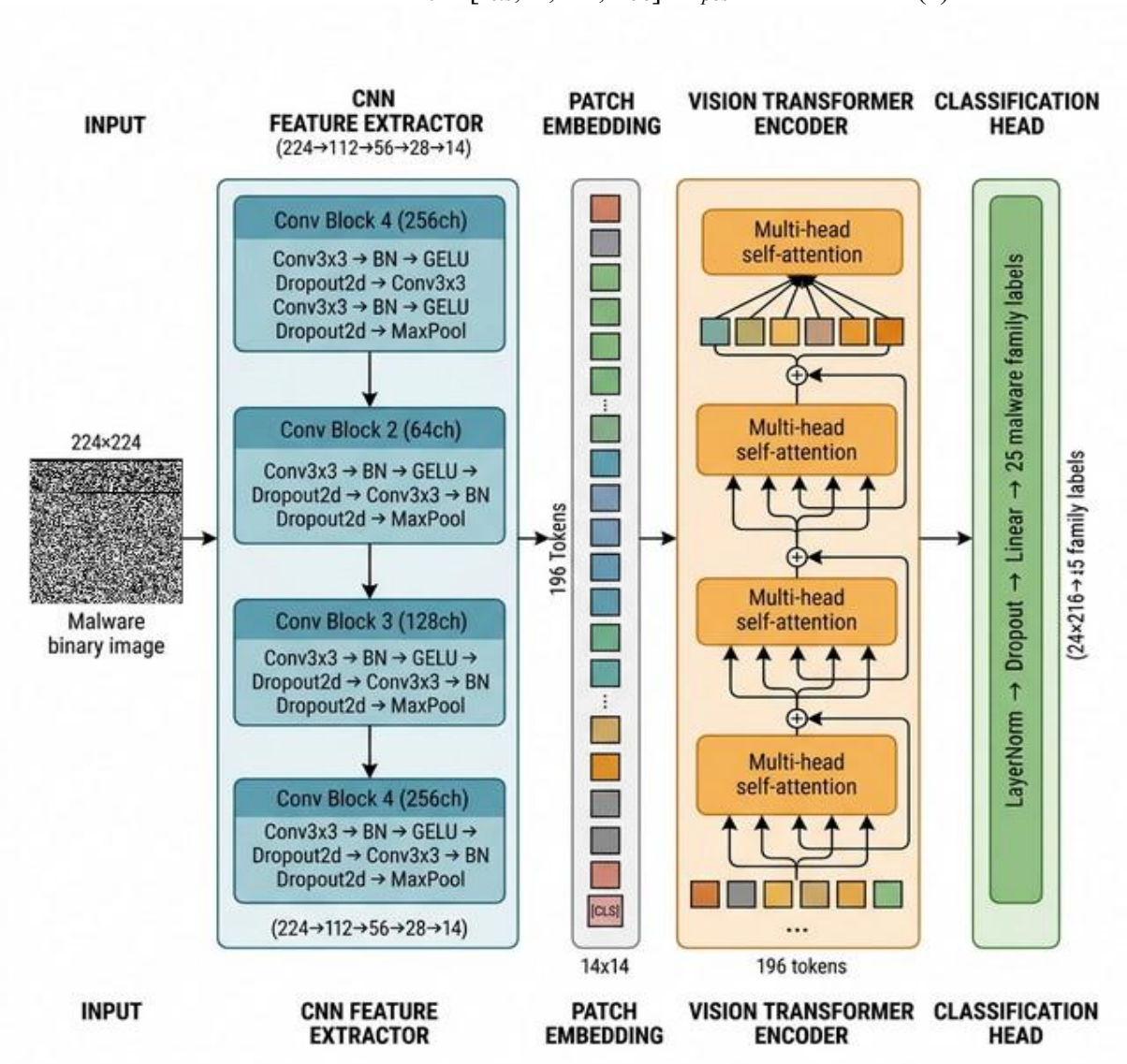

Figure 1: An outline of the suggested Hybrid CNN-ViT architecture

The series is fed through $L = 4$ Transformer encoder layers, all of them containing multi-head self-attention (MHSA) with $H = 8$ heads and a feed-forward network (FFN):

$$Z'_\ell = \mathrm{MHSA}(\mathrm{LN}(Z_{\ell-1})) + Z_{\ell-1} \quad (3)$$

$$Z_\ell = \mathrm{FFN}(\mathrm{LN}(Z'_\ell)) + Z'_\ell \quad (4)$$

The self-attention mechanism calculates:

$$\mathrm{Attentions}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \quad (5)$$

where $d_k = 256/8 = 32$ is the per-head dimension. This enables patches to serve all other patches, and grab world-localised corruption-resistant structural dependencies.

3.4. Classification Head and Focal Loss

The time series representation $z^{(L)}$ is [CLS] and [CLS]. Passed *cls* R^{256}. with Layer Normalization, dropout ($p = 0.3$), and linear layer to generate class logits. this model uses a Focal Loss training [10]. and 2.0 is used in preference to 1/frequency class weights to address the harsh class division in Malimg (family sizes go between 80 to 2,949 samples):

$$\mathrm{FL}(p_t) = -\alpha_t(1 - p_t)^\gamma \log(p_t) \quad (6)$$

Focal Loss downweights well-classified (easy) examples and centres the gradient change on samples belonging to minority classes that are difficult to learn. This is a direct solution to the Autorun.K misclassification problem where the model has never foresaw this 80-sample minority class.

3.5. Aggressive Training Augmentation

To additional precondition the model with obscuration scenarios, this model uses aggressive augmentation of training time: Random Horizontal Flips ($p = 0.5$), Random Rotation (15), Random Affine translation (8 percent), GaussianBlur (kernel size 5, σ [0.1, 2.0]), and RandomErasing ($p = 0.5$, scale up to one third of the area of the image). This compels this model to learn some corrupted images even at the stage of training.

4. Experiment

4.1. Dataset

The model is empirically tested on Malimg dataset [3]. benchmark set of 9,339 malware binary images in grayscale. malware family ages that are 25 in number. Each image is created through a transformation of the raw byte state of a malware. can be executed on a two-dimensional pixel array. The dataset has great class imbalance as in sizes of families. from 80 samples (Yuner.A) to 2,949 samples (Allaple.A). Figure 2 presents a visualisation of class distribution. The dataset is pre- divided into training (8,404 samples, 80 155), and validation (935). samples, ~ 10%.

4.2. Training Configuration

Table 1 summarizes the training hyperparameters. This setup employs mixed FP16 training (mixed precision) with PyTorch Automatic. To optimise memory usage on the, Mixed

Precision (AMP) will be used. 3060 NVIDIA GeForce (6 GB VRAM). All experiments are conducted with CUDA 13.2 and PyTorch 2.x.

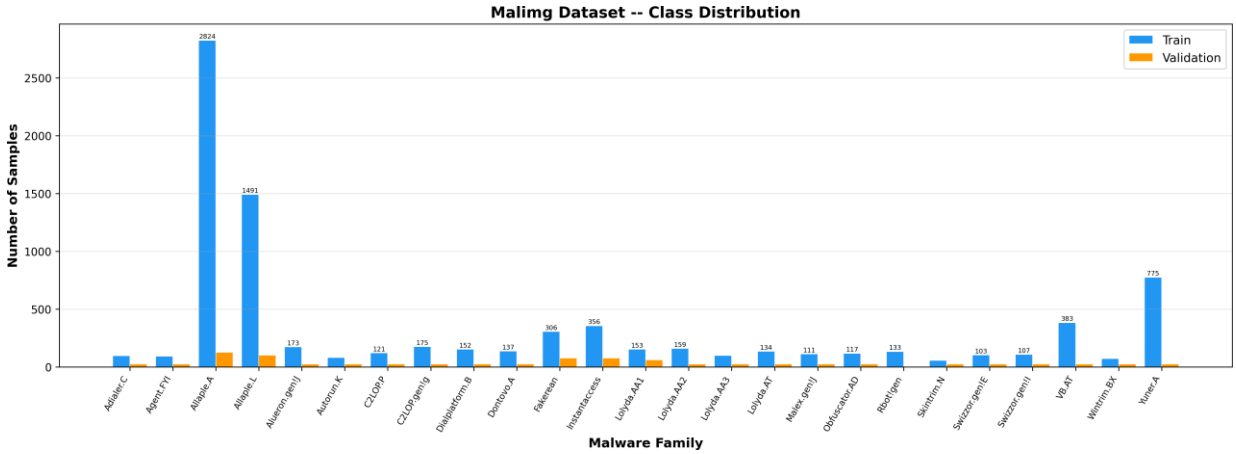

Figure 2: Class distribution of the Malimg dataset across 25 malware families, showing the train/validation split.

Table 1: Training Hyperparameters

Parameter	**Value**
Optimizer	AdamW
Learning Rate	3×10^{-4}
Weight Decay	5×10^{-4}
Batch Size	64
Epochs (max)	20
LR Scheduler	Cosine Annealing
Label Smoothing	0.05
Loss Function	Focal Loss (γ = 2.0) Early Stopping Patience 8 epochs
Gradient Clipping	Max norm 1.0
Precision	Mixed (FP16 via AMP)
CNN Channels	[32, 64, 128, 256]
CNN Spatial Dropout	p = 0.2 ViT Dimension 256
ViT Depth	4 layers
Attention Heads	8
FFN Hidden Dim	512
ViT Dropout	0.15 (encoder), 0.3 (head)

4.3. Baseline Design: Controlled Ablation

To separate the contribution of the ViT, this study designs a CNN baseline controlled ablation: it relies on the same CNNFeatureExtractor (Spatial Dropout None) followed by Adaptive Average Pooling and a linear classifier. It has 1.18M parameters on the baseline, compared to 3.40M for the hybrid. The existing architectural distinction is the existence of the ViT, which makes encoder, patch embedding, and Spatial Dropout a fair comparison that is scientifically rigorous.

5. Result and Discussion

5.1. Clean Dataset Classification Performance

The hybrid model had an overall test accuracy of on the Malimg validation set, it performs with 93.80% and Macro F1 of 0.9080 and a Macro AUC of 0.9980. The CNN baseline on the same clean data demonstrated a 95.40% accuracy. The slightly better clean accuracy of the baseline is anticipated: without the CNN, it can fully leverage the local Spatial Dropout overhead and the ideal (non-adversarial) features of texture. Fig. 3 shows the multi-class AUC ROC curves that indicate confident boundaries in decision making among all 25 classes, even when there is tremendous imbalance of classes.

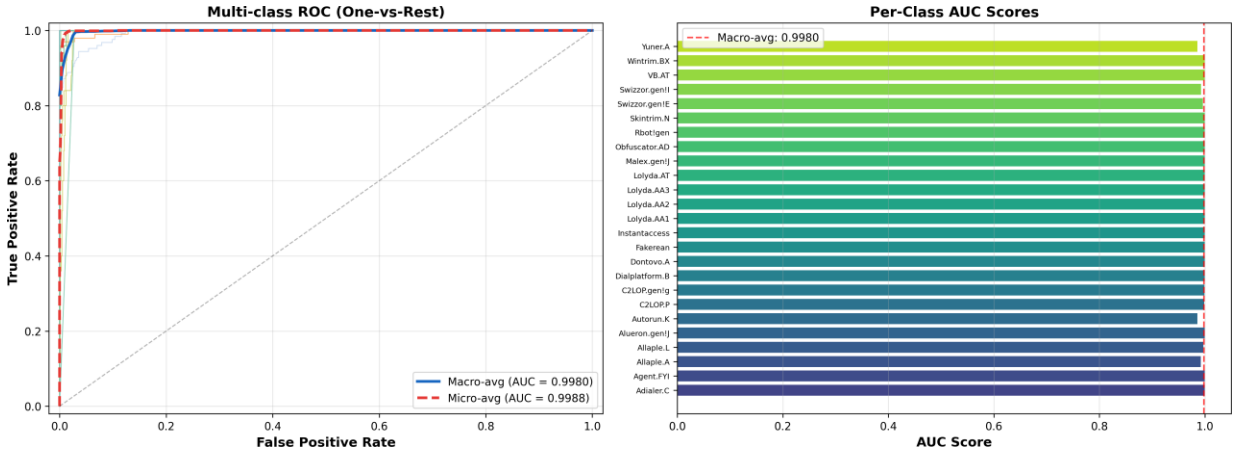

Figure 3: Multi-class ROC curves with Macro-Average AUC of 0.9980, demonstrating robust discrimination across all 25 malware families.

5.2. Obfuscation Robustness: The Core Experiment

To test the central hypothesis, the validation set was subjected to simulated local obfuscation by injecting random byte-level noise at probabilities of 0%, 10%, 20%, 30%, 40%, and 50%. Both models were evaluated on identically corrupted images. Table 2 presents the complete results with the degradation plotted in Fig. 4.

The results are striking. At 0% noise, the CNN baseline leads by 1.60 pp, demonstrating that pure CNNs are marginally superior on clean data where their local texture filters are maximally effective. However, under even moderate obfuscation (10% noise), the hybrid overtakes the baseline, and the gap widens dramatically: at 30% noise, the hybrid retains 81.82% accuracy while the CNN collapses to 52.62%, a gap of 29.20 pp. At 40% noise, the CNN is essentially at random guess level (26.31%), while the hybrid still provides meaningful classification (56.36%).

This crossover pattern CNN leading on clean data but collapsing under noise is the precise signature of the localized texture vulnerability predicted by the theoretical framework of Brosolo et al. [7].

Table 2: Obfuscation Robustness: Hybrid CNN ViT vs. CNN Baseline

Noise (%)	Hybrid CNN-ViT		CNN Baseline		Gap (pp)
	Acc	F1	Acc	F1	
0	93.80	0.908	95.40	0.921	−1.60
10	92.62	0.900	89.73	0.833	+2.89
20	88.77	0.863	77.43	0.662	+11.34
30	81.82	0.811	52.62	0.458	+29.20
40	56.36	0.481	26.31	0.230	+30.05
50	34.65	0.200	12.73	0.088	+21.92

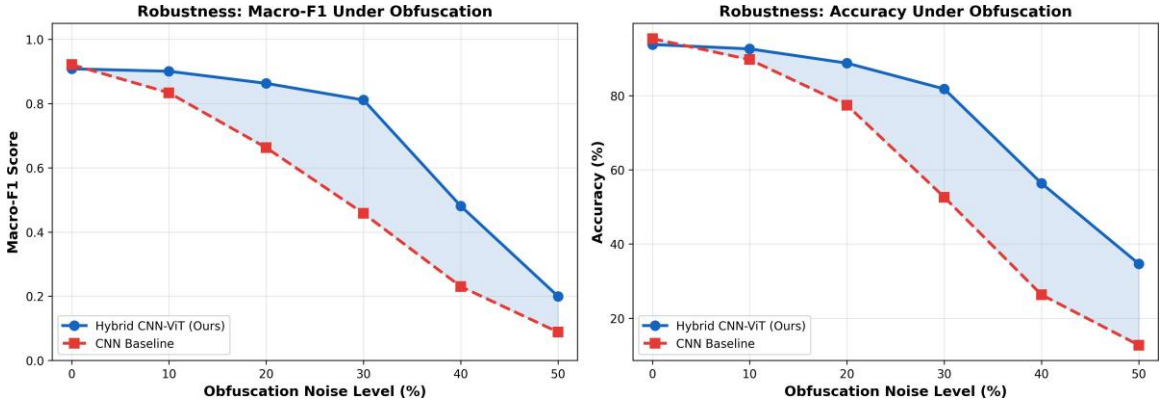

Figure 4: Accuracy and Macro-F1 degradation under simulated local obfuscation. The blue shaded region represents the hybrid's robustness advantage, which peaks at 30 percentage points at 40% noise.

5.3. Confusion Matrix and Per-Class Analysis

The per-class F1 scores and confusion matrix (Fig. 6). The behaviour of the model is provided at all 25 families indicated by (Fig. 5). The Focal Loss technique worked out the *Autorun.K* misclassification: this class improved from 0% F1 score (with cross entropy under normal conditions) to 66.7% F1 (100% recall, 50% precision). The other problematic course is *Yuner.A*, which shares extreme visual similarity with *Autorun.K* due to almost identical binary structures: a well-known weakness of visualisation-based frameworks of similar malware variants. The model attains an ideal F1 score (1.0) on 12 of the 25 families.

6. Explainability Analysis: Visualizing Robustness

This section is the main scholarly input of this paper. Following the explainability paradigm of Brosolo et al. [7] and Asam et al. [8], metrics are not reported; instead, visual evidence is

provided to explain why the hybrid architecture is resistant to obfuscation.

6.1. Methodology

The final averaged channels of the activations were removed and used to form the CNN baseline as a proxy for spatial focus (analogous to the GradCAM model employed by [8]). Simultaneously, the mechanism removed the Multi-Head Self-Attention matrix from the final Transformer layer, and the [CLS] token's attention was isolated from the 196 image patches. These maps were upsampled and superimposed on clean and obfuscated (30% byte noise) copies of prototypical malware code.

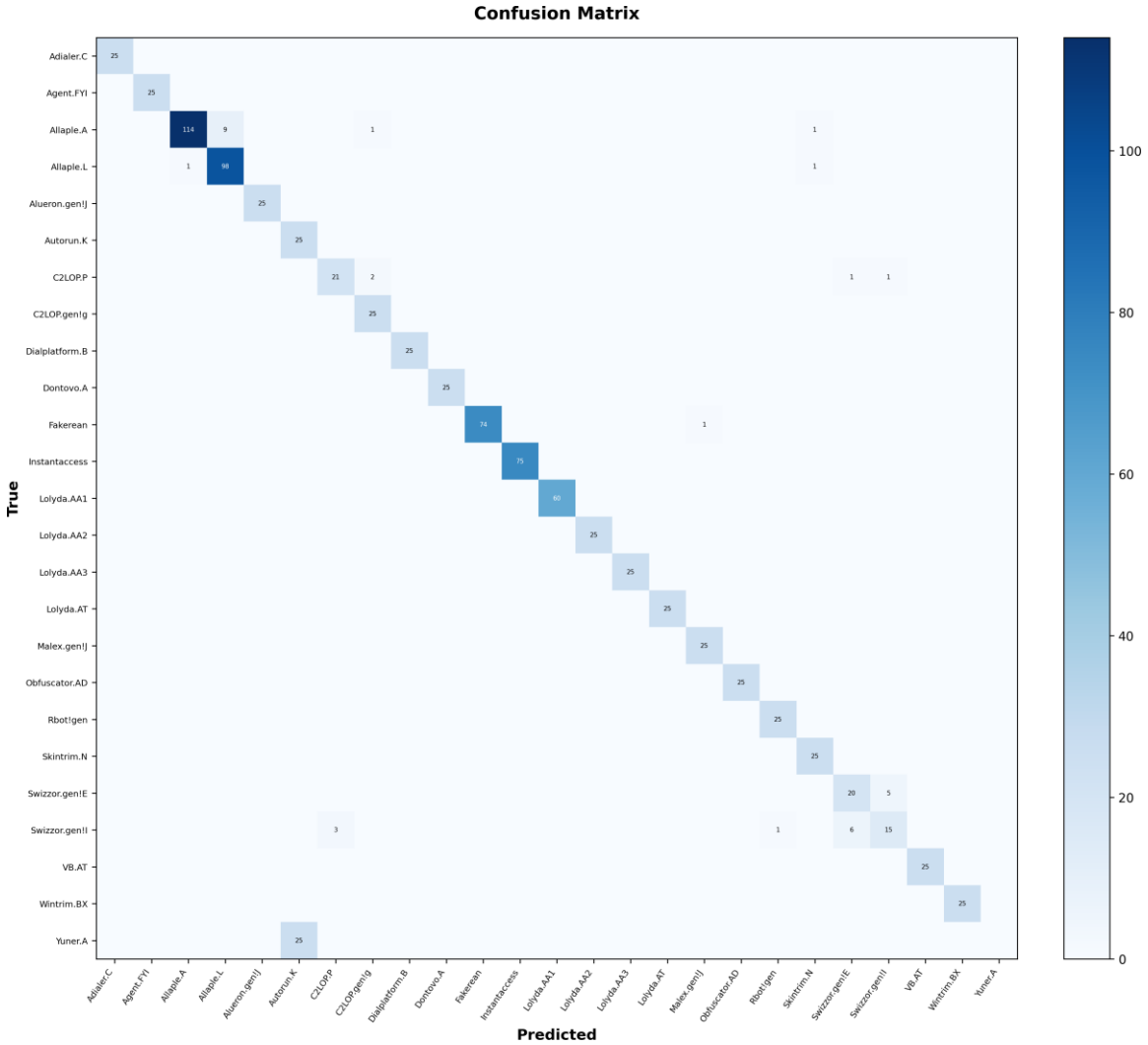

Figure 5: Confusion matrix for the Hybrid CNN-ViT on the Malimg validation set. Focal Loss resolves the Autorun.K zero-classification problem.

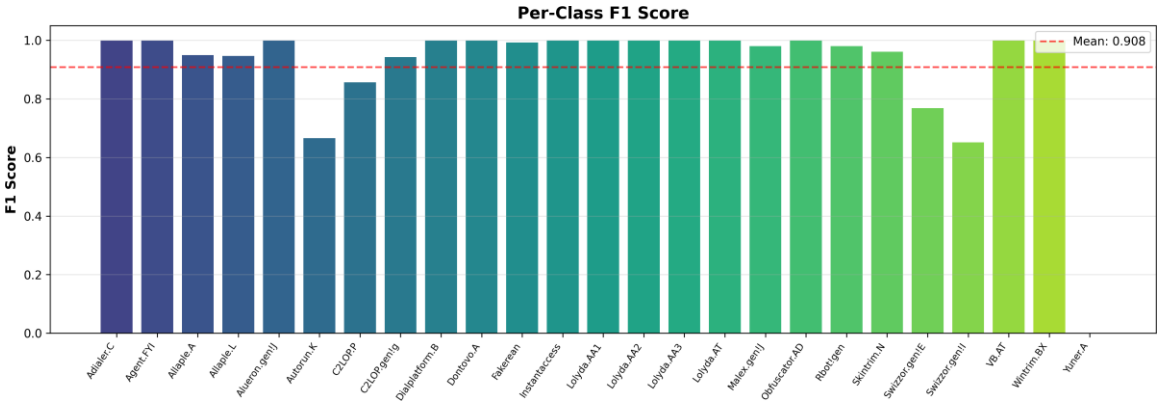

Figure 6: Per-class F1-scores across 25 malware families. 12 classes achieve perfect F1=1.0.

6.2. Comparative Attention Results

Fig. 7 has the graphical validation of the thesis statement. On the CNN activation map, as well as the ViT, are clean images lock attention onto similar distinctive structural parts, the header blocks, import tables and code sections that characterize all families of malware. The two models are similar in performance, as local textures are preserved.

But when 30% noise is introduced into the local bytes, the CNNs cannot identify the systems, and the activation map becomes violently dispersed. The CNN is preoccupied with the turbulent local gradients of the noise signature, diffusing its concern over corrupted areas between the malicious and non-malicious payload. In comparison, the self-attention matrix of the ViT dynamically refocuses. Since every patch is attended to all 196 patches at the same time (through Eq. 5), the Transformer is aware that the noise is not part of the larger structural sequence and shifts its attention to stick tenaciously to the travelling global bits of the payload.

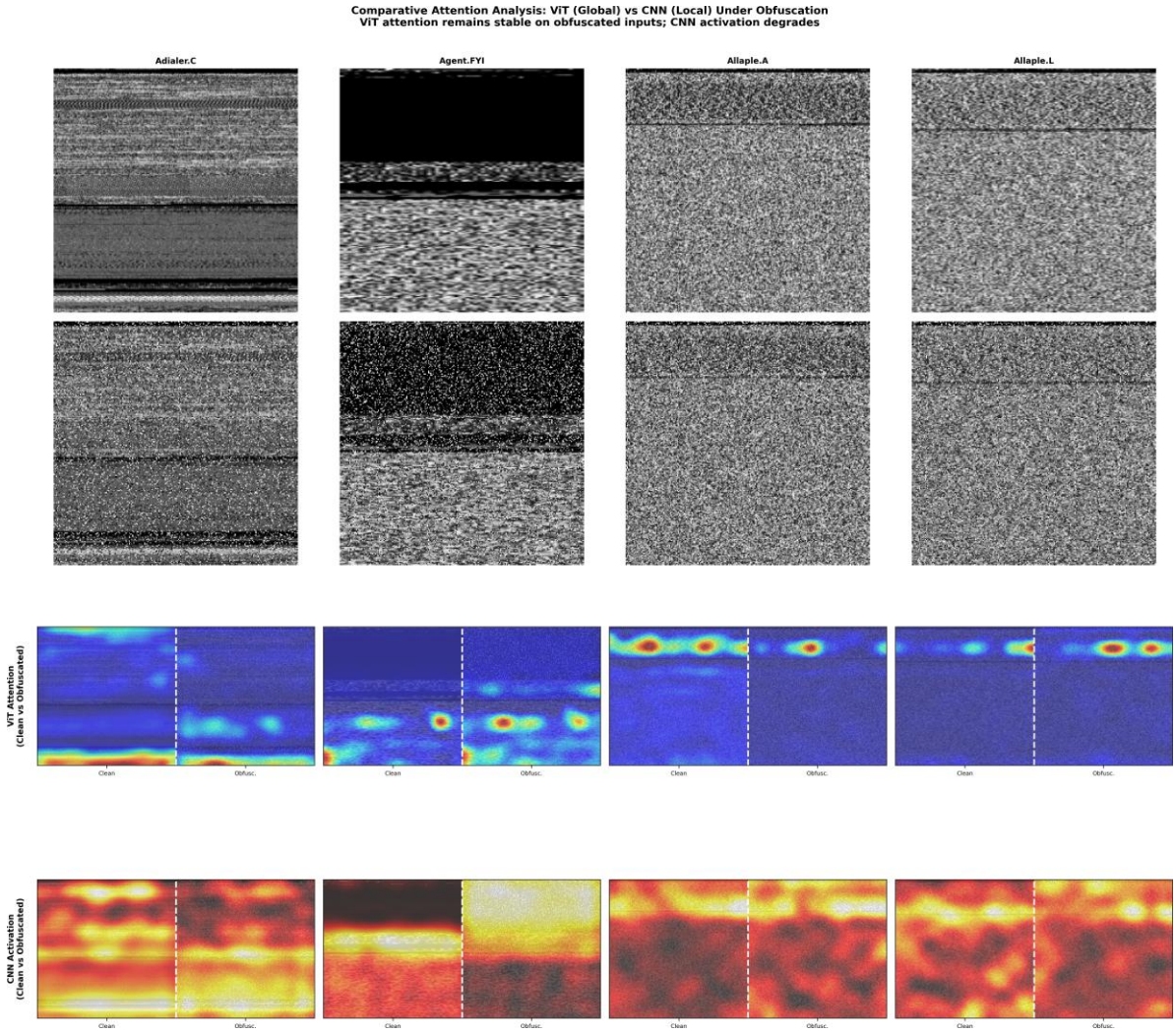

Figure 7: Comparative Attention Analysis under 30% obfuscation. Top rows: clean vs. obfuscated mal- ware. Bottom rows: ViT attention remains focused on global structural payloads while CNN activations degrade and scatter across noise artifacts. This explains the 29.20 pp accuracy gap at 30% noise.

6.3. Theoretical Interpretation

The attention refocusing behaviour is a mathematically related explanation. There are no factors for the self-attention calculation (Eq. 5). The contribution made by the patch token to the output of [CLS] is weighted with the softmax-normalised dot product similarity. When noise evils a part of patches, their feature representations get random, giving poor scores in similarity to the [CLS] query. The normalisation is then used to apply softmax automatically and redistributes the attention weight to the rest, which are not corrupted patches. This is implicit denoising of an archi type, evidently lacking in CNNs, in which the receptive field of each neuron is identified as a field with a fixed kernel width that cannot "skip over" corrupted localities.

7. Conclusion

The vulnerability that is so critical to the vision-based approach is covered in this paper,

malware classification: the vulnerability of CNN feature ex- obfuscation to localised obfuscation. The suggestion was a CNN hybrid ViT model with Focal and Spatial Dropout Loss, which shows that a pure CNN base collapses from 95.40% to 52.62% under 30% byte level noise, the hybrid model crashes gracefully with an error
93.80−81.82 = 0.68 above 93.80% to 81.82% = 0 point). The advantage of robustness of 29.20%.
More to the point, this paper offers elucidable proof of this robustness. The heat maps of comparison
visualise that the ViT has a global self-attention mechanism that is naturally able to disregard local noises dispersed about and dynamically re-point to unremitting, worldwide dispersed malware structural payloads. This explainability dimension, prevalent in modern literature, contributes to going beyond standards of accuracy in architecture.

The controlled ablation (identical CNN backbone), curves of quantitative robustness across six noise levels, and a graphical XAI demonstration provide a strict framework for assessing malware classifiers in adversarial settings. This assessment will advance to real-world malware-packed samples, deploy the model as a real-time detector pipeline, research opponent training strategies in more detail, and harden.

References

[1] "AV-TEST – the independent IT security institute: Malware statistics," https://www.av-test.org/en/ statistics/malware/, 2024, accessed: 2024-12-01.

[2] D. Ucci, L. Aniello, and R. Baldoni, "Survey of machine learning techniques for malware analysis," *Computers & Security*, vol. 81, pp. 123–147, 2019.

[3] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware images: Visualization and automatic classification," *Proceedings of the International Symposium on Visualization for Cyber Security (VizSec)*, pp. 1–7, 2011.

[4] M. Kalash, M. Rochan, N. Mohammed, N. D. B. Bruce, Y. Wang, and F. Iqbal, "Malware classification with deep convolutional neural networks," in *Proceedings of the 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*. IEEE, 2018, pp. 1–5.

[5] D. Vasan, M. Alazab, S. Wassan, B. Safaei, and Q. Zheng, "Image-based malware classification using ensemble of CNN architectures (IMCFN)," *Computers & Security*, vol. 92, p. 101748, 2020.

[6] D. Gibert, C. Mateu, and J. Planes, "The rise of machine learning for detection and classification of malware: Research developments, trends and challenges," *Journal of Network and Computer Applications*, vol. 153, p. 102526, 2020.

[7] M. Brosolo, P. Vinod, and M. Conti, "The road less traveled: Investigating robustness and explainability in CNN malware detection," *arXiv preprint arXiv:2503.01684*, 2025.

[8] M. Asam, S. H. Khan, A. Akbar, S. Bibi, T. Jamal, and A. Khan, "An explainable visual malware classification framework using convnextswin hybrid architecture," *Information Sciences*, vol. 661,
p. 120200, 2024.

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani,

M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[10] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980– 2988.

[11] K. S. Han, J. H. Lim, B. Kang, and E. G. Im, "Malware analysis using visualized images and entropy graphs," in *International Journal of Information Security*, vol. 14, no. 1. Springer, 2015,
pp. 1–14.

[12] Z. Dai, H. Liu, Q. V. Le, and M. Tan, "CoAtNet: Marrying convolution and attention for all data sizes," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 34, pp. 3965–3977, 2021.

[13] F. U. Khan, S. Aziz, and N. Iqbal, "LeViT-MC: A lightweight vision transformer for malware classification," *arXiv preprint arXiv:2402.16995*, 2024.