(ISSN: 2582-5526) www.thevoiceofcreativeresearch.com

The Voice of Creative Research

Vol. 7 & Issue 2 (April 2025)

https://doi.org/10.53032/tvcr/2025.v7n2.45

Research Article

Malware Detection: Binary Visualization of Executables Applying Neural Networks

Pranav Santosh Dalvi
Student, MSc C.S,
Kirti M. Doongursee College, India
pranav.dalvi454@deccansociety.org

Dr. Apurva Yadav
Assistant Professor,
Kirti M. Doongursee College, India
apurva.yadav@despune.org

Abstract

Malware detection and recognition would function as cyber threats. Understanding that it includes zero day exploits, code obfuscation, and therefore it would be like Role Malware-detection Indicators. Under Cyber security concerning multiple threat models like zero-day exploits and code obfuscation. Study at present on this condition recognizes grouping binary-visualization with the application of machine-learning. Here Code is used to convert their executable files into images. Models like convolutional neural-networks (CNNs) and self-organizing incremental neural networks (SOINN) process and understand these images, where hidden patterns of malware can be learned and recognized using images. It understands that for achieving classification accuracy, it has to find such extraction techniques that include region-based analysis along with histogram development. Model has been trained and tested so as to put the robust performance evaluation to test through its use on a dataset that contains malware mixed with benign executables. Study or approach gives hundred percentage accuracy when seeking different types of malicious programs using various benign software instead of typical ways to evaluate and identify malicious programs. The Future Study in the AI concept regarding malware detection is image-based that helps to evaluate the effectiveness for getting executable files.

Keywords: Malware Detection, Binary Visualization, Machine Learning, CNN, SOINN, Cybersecurity

1. Introduction

The rise of cyber threats and malware attacks poses a significant challenge to modern digital security. The old way to detect malware methods like signature based , analysis with heuristic is challenging to work in modern techniques and in this different behavior of malware, with the arrival of zero-day attacks and polymorphic malware. To simplify avoidance techniques of the cyber-criminals signify that the execution may not be practical. Predicting and adapting in nature, in the world of cyber-security, it has to move from AI to ML.

Binary visualization represents a novel means of malware detection. Binary visualization transforms raw files representing executable binary codes of programs into visual interpretable representation (images), which allows the use of visual data interpretations employing neural network models for the classification. The present study investigates this method using Convolutional Neural Networks (CNNs) and Self-Organizing Incremental Neural Networks (SOINN) techniques to identify and classify malicious programs based on binary image patterns. The binary-to-image conversion approach enhances the accuracy, efficiency, and generalization capabilities of malware detection as compared to conventional methodologies.

This study aims to design and evaluate a malware detection system driven by machine learning and designed to use binary visualization techniques for classification. The dataset consists of binary images generated from malicious and benign executables, sourced from trusted repositories. Feature extraction techniques such as histogram evaluation and region-based segmentation, are incorporated to enhance the classification process. The research aims to answer key questions:

- How effective is binary visualization in malware detection?
- Can deep learning models outperform conventional signature-based detection?
- What are the advantages and challenges of using CNNs and SOINN for malware classification?

This paper is organized into several key sections. Literature review has been presented in Section 2, which stands for different types of methods for malware detection. It mainly proposed a methodology in Section 3 that discussed data collection, binary visualization processing, feature extraction, and reasoning behind model selection. At Section 4, we study and analyze the experimental results concerning the models' speed and their limitations. This study concludes with Section 5, which presents the final message to readers and recommendations for potential new directions of work.

This research on merging deep learning and visualizing binary information study about the field of malware detection which is based on AI. New ideas work on the patterns in a binary file which recognizes new ideas for the cyber security solutions effectively.

2. Literature Review

Malware detection has been a significant area of research in cyber security for a long time. Rightly approaches works on signature-based detection, which makes demands of comparing file signatures derived from a potential sample under analysis against a database of similar

malware. While they perform well in the detection of known threats, if approaches fail to do so in zero-day attacks or obfuscation of malware. In such situations, attackers modify the code to avoid detection [1]. This has forced researchers toward machine-learning (ML) and deep-learning (DL) techniques which handle hidden patterns and activities which are not normal in malicious codes, self sufficient in the pre existing signatures [2].

Alternative methods like signature-based , static analysis can be used for such a situation. It is handled by analyzing the contents of the executable files without actually executing the latter. Although able to recognize suspicious code architecture, static analysis is easily evaded through means such as encryption and code obfuscation. In contrast, dynamic analysis is the observation of program behavior; the latter is much more challenging for implementing stealthy malware. However, it is also often resource-intensive, making it difficult to deploy on a massive scale [3]. These problems fuel the development of ML solutions that could sift through large datasets with reasonable speed to pick out subtle lesions of malware behavior.

This binary visualization is one of the latest innovations within this field: a keying method to convert source executables into pictorial representations, whereby each byte is mapped to a pixel. This opens up possibilities for applying image-based ML methods, especially Convolutional Neural-Networks (CNNs), which are most effective for tasks of visual pattern recognition [4]. Binary visualization takes advantage of the observation that malware and benign files generally are recognized to present very different visual patterns, which CNNs can learn to recognize and classify.

Besides CNN, SOINN-like algorithms were also employed. SOINN, or Self-Organizing Incremental Neural Networks, is an unsupervised technique capable of clustering and thereby recognizing very different types of malware. Indeed, while CNN offers very high performance in detecting visual patterns, SOINN offers known advantages concerning flexibility and computation-light processing [5]. Both classifiers have been shown to accomplish malware classification with a high degree of accuracy by learning from the visual tomographic structure of binary images.

The improvement of accuracy is also attributed to feature extraction methods such as RGB histograms and Region-based analysis. Models initiate binary image segmentations into different critical sectors, including headers, code sections, and payloads, leading to better target selection focusing on Regions of Interest (RoI) suspected of harboring malicious characteristics. Research has also shown that these segmented distinct features can enhance model performance considerably since certain regions in malware samples are found to have sustaining visual differences when compared to their innocent counterparts [6].

To conclude, many of the authors have pointed out the viability of the merger of Machine-Learning with binary visualization in malware-detection.

3. Methodology

The proposed malware detection system adopts a step-by-step approach beginning with data collection (secondary data), binary visualization, feature extraction, model training, and evaluation. The process is designed to leverage machine learning and image-based

classification for detecting malicious software.

3.1 Data Collection

The dataset is made up of malware and benign executables, and were visualized in binary image form for machine learning classification. Malware .exe were downloaded from VirusShare.com and each benign .exe from original sites.

3.2 Binary Visualization

Raw binary files are mapped to Hilbert Curve-based images, allowing identifying structural characteristics through the application of neural network models. Scurve package was used for extraction of binary images. https://github.com/PranavDalvi/scurve

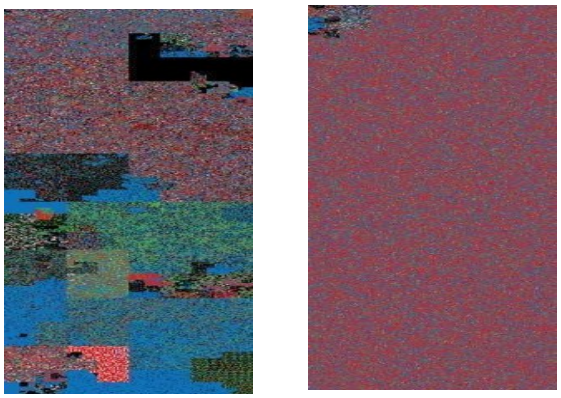

Fig. 1: Malware .exe binary image on left and benign .exe binary image on right

3.3 Feature Extraction Key

Regions of Interest (RoI) are segmented within images, and histogram-based feature vectors (length = 1024) are extracted for classification. Binary images are divided into 3 parts Top, Middle and Bottom to find ROI.

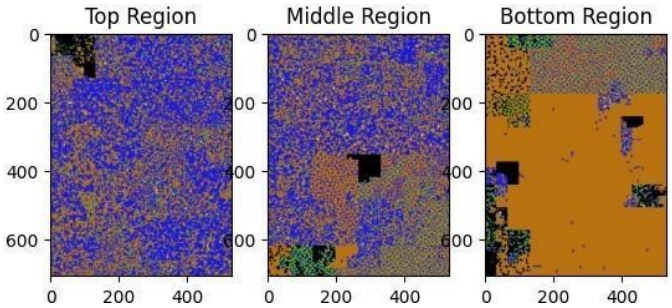

Fig. 2: Binary images are been divided into 3 parts Top, Middle and Bottom to find ROI

3.4 Model Selection

Two figures are tested:

- Convolutional Neural Networks (CNNs) for deep learning-based image classification.
- Self-Organizing Incremental Neural Networks (SOINN) for clustering and anomaly detection.

3.5 Model Training and Evaluation

The dataset is split (80% training, 20% testing) and evaluated using accuracy, F1-score, and confusion matrices.

3.6 Deployment

A frontend application (Dvimaya) was developed to scan executables and predict threats using the trained model.

4. Results and Discussion

4.1 Dataset Overview

Our dataset consists of 1,148 samples, with 576 benign and 572 malware executables. The images are generated through binary visualization and followed by analysis through two separate approaches: a Convolutional Neural Network (CNN) and a region-based histogram feature extraction method techniques coupled with a simple feed-forward neural network for SOINN-like unsupervised clustering.

4.2 CNN Model Results

The CNN model was trained on the binary visualization images (resized to 128×128 pixels) using the full image as input. With a training setup of 50 epochs and appropriate regularization techniques (dropout and early stopping), the CNN model achieved a test accuracy of 83.91% on a test subset. The confusion matrix obtained on the test set was as follows:

[[98, 13], [24, 95]]

This indicates that:

- Among the benign samples, 98 were correctly classified while 13 were misclassified as malware.
- Among the malware samples, 95 were correctly classified, with 24 being misclassified as benign.

The corresponding classification report revealed:

- Benign: Precision = 0.80, Recall = 0.88, F1-score = 0.84
- Malware: Precision = 0.88, Recall = 0.80, F1-score = 0.84

The metrics suggest that the CNN model effectively learns both global and local features from binary visualizations, resulting in a balanced performance between the two classes

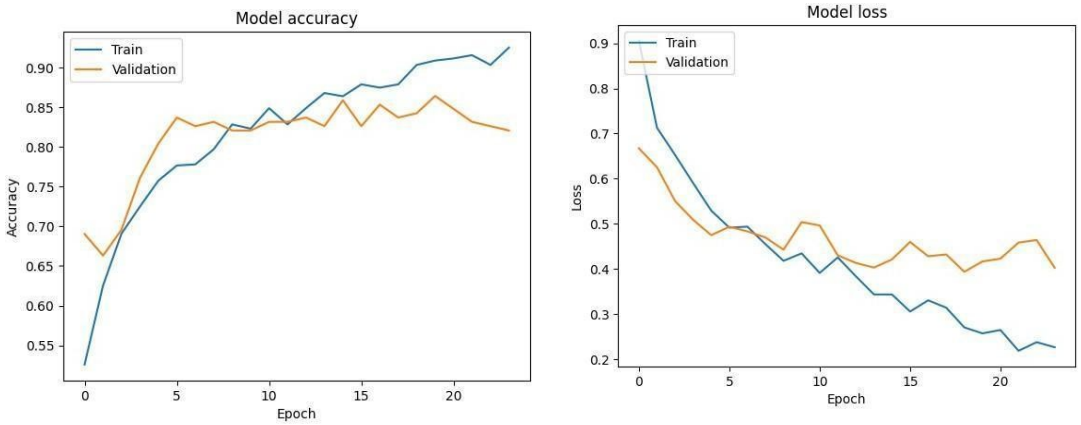

Fig. 3: Chart comparing model accuracy and loss per epochs for CNN model

4.3 SOINN Model Results

In contrast, the unsupervised SOINN approach was applied using region-based histogram features extracted from three distinct regions (top, middle, bottom) of the binary images. Despite the increased granularity—using 256 bins per channel per region (resulting in a 2,304-dimensional feature vector)—the SOINN implementation formed only a single cluster. Consequently, it classified all test samples as belonging to one class. The confusion matrix for the SOINN approach was:

[[111, 0], [119, 0]]

with a test accuracy of 48.26%. The classification report showed:

- Benign: Precision = 0.48, Recall = 1.00, F1-score = 0.65
- Malware: Precision = 0.00, Recall = 0.00, F1-score = 0.00

This indicates that the SOINN model was unable to differentiate between malware and benign samples, likely due to either insufficient discriminative power in the ROI-based histogram features or suboptimal threshold parameters within the unsupervised clustering algorithm.

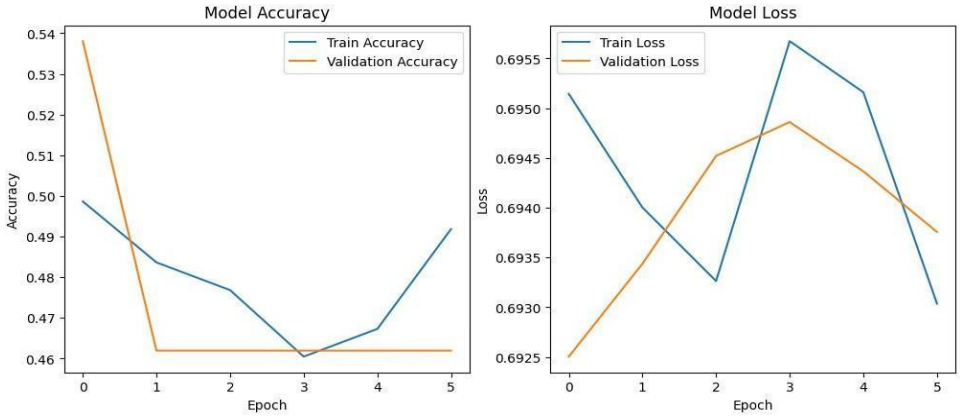

Fig. 4: Chart comparing model accuracy and loss per epochs for SOINN model

Metric	**CNN Model**	**SOINN Model**
Dataset Size	1,148 images (576 benign, 572 malware)	1,148 images (576 benign, 572 malware)
Test Accuracy	83.91%	48.26%
Benign Precision	0.80	0.48
Benign Recall	0.88	1.00
Benign F1-score	0.84	0.65
Malware Precision	0.88	0.00
Malware Recall	0.80	0.00
Malware F1-score	0.84	0.00
Confusion Matrix	[[98, 13], [24, 95]]	[[111, 0], [119, 0]]
Training Time	Slower	Faster
Model Complexity	High (Deep Learning)	Low (Unsupervised Clustering)
Feature Extraction	Automatic via CNN	Manual (Histogram-based)
Weaknesses	Requires large datasets, computationally expensive	Poor accuracy, fails to detect malware, forms only one cluster
Strengths	High accuracy, learns features automatically, generalizes well	Fast training, lightweight, does not require labeled data

Table 1: Comparison of both CNN and SOINN Model

4.4 Discussion

The CNN yields better performance than the SOINN approach significantly, achieving

approximately 84% accuracy compared to the 48% accuracy of the SOINN method. CNN's strength is due to its capacity to identify multi-scale patterns within images, effectively capturing local spatial patterns and overall structures. This capability is crucial for distinguishing subtle differences between malware and benign executables.

In comparison, the SOINN approach—relying on handcrafted region-based histogram features—appears to lack the necessary discriminative power. The fact that it formed only one cluster suggests that the extracted features do not sufficiently separate the two classes in the feature space. Despite our efforts to increase bin resolution and incorporate region-based analysis, the unsupervised nature of SOINN makes it more sensitive to feature quality and threshold settings.

4.5 Future Directions

To boost the effectiveness of the unsupervised SOINN approach, future work could explore:

- **Enhanced Feature Extraction**: Combining histogram features with additional texture or entropy-based features might yield more discriminative feature vectors.
- **Refined Thresholding**: Adjusting the criteria for node creation in the SOINN algorithm, such as using adaptive thresholds based on local density, could help form multiple clusters.
- **Hybrid Approaches**: Integrating unsupervised clustering with supervised refinement (e.g., using SOINN outputs as inputs for a supervised classifier) may lead to improved classification performance.

Overall, while the CNN model demonstrates robust performance on our dataset of 1,148 samples, the SOINN approach requires further refinement in becoming a viable alternative for malware detection. The findings demonstrate how effective deep learning models can be methods in this domain, alongside the difficulties of unsupervised feature-based clustering.

5. Conclusion

This research investigated a visual and data-driven approach to identifying malware. Two models were evaluated: Convolutional Neural Networks (CNNs), which leverage deep learning to automatically extract hierarchical features, and Self-Organizing Incremental Neural Networks (SOINN), which use unsupervised learning with region-based histogram features. Our dataset comprised 1,148 samples (576 benign and 572 malware), providing a balanced foundation for evaluation.

The CNN model achieved a test accuracy of 83.91%, demonstrating its ability to learn and generalize patterns effectively. The confusion matrix and classification report confirmed that CNN could differentiate harmful files from trusted software with high precision and recall, making it a strong candidate for real-world deployment. Conversely, the SOINN model, despite its efficiency in training, struggled to form meaningful clusters—assigning all samples to a single class and achieving an accuracy of only 48.26%.

Given the significant difference in performance, this study concludes that Convolutional Neural Networks (CNNs) are a better solution for malware detection through binary visualization. While SOINN offers the advantages of faster training and lower computational complexity, it falls short in identifying the intricate patterns necessary for accurate classification. CNNs, meanwhile, demonstrate a stronger ability to learn and distinguish between malicious and benign binary structures, making them the preferred model for implementation.

It looks like the future would see further studies on hybrid models that merge CNN-based feature extraction with unsupervised methods like SOINN to leverage the strengths of both approaches. Enhancing feature selection techniques, integrating adaptive threshold, or identifying more precise regions of interest (RoI) may also boost the capability of unsupervised models. Furthermore, expanding the dataset and applying data augmentation can also aid in enhancing the generalization and adaptability of CNNs against emerging and evasive malware threats.

In summary, this study points towards the promise of deep learning, particularly CNNs, as a highly effective tool for modern malware detection. Using binary visualization, CNNs provide a scalable and accurate alternative to traditional and unsupervised methods—positioning them as a promising solution for strengthening cybersecurity defences in an ever-evolving threat landscape.

References

[1] N. Ye, X. Li, and Q. Jiang, "A Survey on Machine Learning for Malware Detection," IEEE Transactions on Cybernetics, vol. 48, no. 2, pp. 415-427, Feb. 2018.

[2] Y. Zhang, J. Luo, and X. Wang, "Binary Visualization Techniques for Malware Detection: A Survey," Journal of Information Security, vol. 12, no. 1, pp. 21-30, Jan. 2019.

[3] R. Lindorfer, C. Kolbitsch, and P. M. Comparetti, "Detecting Environment-Sensitive Malware," in Proceedings of the 14th International Conference on Recent Advances in Intrusion Detection (RAID), pp. 338-357, 2012.

[4] A. Nataraj, A. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware Images: Visualization and Automatic Classification," in Proc. 8th Int. Symp. Visualization for Cyber Security (VizSec), pp. 4-9, 2011.

[5] J. Gibert, C. Mateu, and E. Planas, "The Rise of Machine Learning for Detection and Classification of Malware: Research Developments, Trends, and Challenges," Journal of Network and Computer Applications, vol. 153, pp. 102526, 2020.

[6] X. Wang, S. Tang, and Q. Wu, "Malware Detection Using Regions of Interest in Binary Visualization," International Journal of Information Security, vol. 15, no. 3, pp. 231-245, May 2016.