Cybersecurity and Law

2025 Nr 2 (14)

DOI: 10.34567/cal/215977

Visualisation of binary files and machine learning methods in the detection and classification of malicious software

Wizualizacja plików binarnych i metody uczenia maszynowego w detekcji i klasyfikacji złośliwego oprogramowania

Katarzyna Hanna KAMIŃSKA
Politechnika Warszawa
ORCID: 0000-0001-5726-6967
E-mail: katarzyna.kaminska@pw.edu.pl

Magdalena CEBULA
Politechnika Warszawska
ORCID: 0009-0000-3361-5835
E-maila: magdalena.cebula@pw.edu.pl

Abstract

In recent years, there has been a significant development of methods for analyzing malicious software using machine learning techniques, which have become essential in the face of the growing complexity of modern threats. Traditional approaches based on signature analysis or feature extraction from source code are becoming increasingly ineffective when confronted with obfuscation, polymorphism, and encryption techniques employed by creators of malicious software.

In response to these challenges, the concept of visualizing binary files has emerged, involving the transformation of their byte structure into images that can subsequently be analyzed using image-processing algorithms and machine learning techniques.

This approach enables the identification of patterns and dependencies that are difficult to capture using traditional methods, while also allowing solutions developed in the field of computer vision to be adapted to the analysis of malicious software.

The article presents an overview of the most important visualization methods, including grayscale imaging, color representations, and entropy-based visualizations, and discusses their applications in detection and classification

tasks. Particular attention is given to machine learning models used for analyzing images derived from binary files.

The concluding section highlights both the main advantages of this approach - such as resistance to selected forms of obfuscation, the ability to automatically extract features, and scalability - and its key limitations, including the loss of semantic information and the susceptibility of models to data variability. The article also outlines potential directions for further research, including the development of multimodal methods and continual learning techniques that may enhance the effectiveness and robustness of visualization-based detection systems.

Keywords

malicious software, binary visualization, deep learning, image-based classification, malware detection

Streszczenie

W ostatnich latach obserwuje się intensywny rozwój metod analizy złośliwego oprogramowania z wykorzystaniem technik uczenia maszynowego, które stają się niezbędne wobec rosnącej złożoności współczesnych zagrożeń. Tradycyjne podejścia, oparte na analizie sygnatur lub ekstrakcji cech z kodu źródłowego, okazują się coraz mniej skuteczne w obliczu technik obfuskacji, polimorfizmu i szyfrowania stosowanych przez twórców złośliwego oprogramowania.

W odpowiedzi na te wyzwania pojawiła się koncepcja wizualizacji plików binarnych, polegająca na przekształcaniu ich struktury bajtowej w obrazy, które następnie mogą być analizowane za pomocą algorytmów przetwarzania obrazu i technik uczenia maszynowego. Podejście to umożliwia identyfikację wzorców i zależności trudnych do uchwycenia metodami tradycyjnymi, a jednocześnie pozwala przenosić na grunt analizy złośliwego oprogramowania rozwiązania wypracowane w dziedzinie komputerowego rozpoznawania obrazów.

W artykule przedstawiono przegląd najważniejszych metod wizualizacji, obejmujących obrazowanie w skali szarości, reprezentacje kolorowe oraz wizualizacje oparte na entropii, a także omówiono ich zastosowania w zadaniach detekcji i klasyfikacji. Szczególną uwagę poświęcono modelom uczenia maszynowego wykorzystywanym do analizy obrazów pochodzących z plików binarnych.

W części podsumowującej wskazano zarówno główne zalety tego podejścia, takie jak odporność na wybrane formy obfuskacji, możliwość automatycznego wydobywania cech czy skalowalność, jak i jego kluczowe ograniczenia, obejmujące m.in. utratę informacji semantycznych oraz podatność modeli na zmienność danych. Przedstawiono również potencjalne kierunki dalszych badań, w tym rozwój metod multimodalnych i uczenia ciągłego, które mogą zwiększyć skuteczność i odporność systemów detekcji opartych na wizualizacji.

Słowa kluczowe

złośliwe oprogramowanie, wizualizacja plików binarnych, głębokie uczenie, klasyfikacja obrazów, detekcja złośliwego oprogramowania

Wstęp

Złośliwe oprogramowanie (malware), stanowiące jeden z najpoważniejszych problemów współczesnego bezpieczeństwa informatycznego, charakteryzuje się nieustanną ewolucją oraz zdolnością do adaptacji wobec tradycyjnych metod wykrywania. Wczesne rozwiązania detekcyjne, oparte głównie na sygnaturach,

umożliwiały identyfikację znanych rodzin złośliwego oprogramowania poprzez porównywanie skrótów lub fragmentów kodu binarnego z bazą wzorców. Metody te były jednak bezsilne wobec wariantów tworzonych w sposób automatyczny, które różniły się strukturą, lecz zachowywały tożsame funkcje. W miarę jak techniki obfuskacji, szyfrowania i modyfikacji kodu stawały się coraz bardziej wyrafinowane, konieczne okazało się opracowanie metod, które potrafiłyby analizować złośliwe oprogramowanie nie tylko na poziomie składni, lecz także struktury i zachowania.

Zastosowanie metod uczenia maszynowego (Machine Learning, ML) w detekcji złośliwego oprogramowania wniosło nową jakość do analiz bezpieczeństwa. Wprowadzenie automatycznych systemów klasyfikacji pozwoliło na rozpoznawanie wzorców ukrytych w danych binarnych, metadanych plików czy zapisach z monitoringu dynamicznego. Jednakże skuteczność modeli uczenia maszynowego w dużym stopniu zależała od jakości i rodzaju cech wejściowych, które należało ręcznie przeanalizować oraz odpowiednio oczyścić i przygotować. Ta potrzeba manualnej inżynierii cech ograniczała skalowalność i adaptacyjność systemów. W tym kontekście pojawiła się koncepcja wizualizacji plików binarnych, która umożliwia przedstawienie danych w formie obrazów, a następnie wykorzystanie do ich analizy technik głębokiego uczenia (Deep Learning, DL), znanych z dziedziny rozpoznawania obrazów.

Idea wizualizacji złośliwego oprogramowania zakłada, że ciąg bajtów w pliku binarnym może zostać zinterpretowany jako sekwencja wartości intensywności piksela. Przekształcenie to pozwala odwzorować strukturę pliku w postaci obrazu dwuwymiarowego, w którym wzorce wizualne odpowiadają rzeczywistym strukturom kodu, sekcjom nagłówków, danym wykonywalnym lub zasobom. Dzięki temu badacze i systemy automatyczne mogą wykorzystywać zaawansowane metody analizy obrazów do rozróżniania rodzin złośliwego oprogramowania, identyfikowania podobieństw między próbkami i wykrywania nowych wariantów. Koncepcja ta została po raz pierwszy opisana w pracy Nataraj i wsp.[1], w której złośliwe pliki wykonywalne, sklasyfikowane przez firmę Microsoft, zostały przekształcone w obrazy w skali szarości, a następnie analizowane za pomocą klasyfikatorów k-najbliższych sąsiadów (k-Nearest Neighbors, kNN). Wyniki tych badań dowiodły, że nawet prosta reprezentacja wizualna ujawnia charakterystyczne wzorce, które mogą służyć do rozróżniania rodzin złośliwego oprogramowania z wysoką skutecznością.

Wizualizacja jako podejście do analizy złośliwego oprogramowania

Wizualizacja plików binarnych polega na odwzorowaniu ich zawartości w przestrzeni obrazu, w której każdy bajt jest interpretowany jako wartość intensywności piksela lub składnik koloru. W zależności od przyjętej metody można tworzyć obrazy w skali szarości, obrazy kolorowe lub mapy entropii. Proces ten przebiega zazwyczaj w kilku etapach: najpierw odczytywana jest sekwencja bajtów z pliku, następnie wartości te są rozmieszczane w postaci dwuwymiarowej macierzy o określonych wymiarach, po czym wynikowa macierz przekształcana jest w obraz cyfrowy. W niektórych przypadkach stosuje się

[1] L. Nataraj, S. Karthikeyan, G. Jacob, B.S. Manjunath, Malware images: visualization and automatic classification, Proceedings of the 8th International Symposium on Visualization for Cyber Security (VizSec'11) 2011, no.4, s. 1-7, https://doi.org/10.1145/2016904.2016908.

dodatkowe przetwarzanie, takie jak normalizacja danych, wygładzanie lub wycinanie sekcji pliku, aby zapewnić spójność wymiarów obrazu.

Wizualne przedstawienie pliku binarnego pozwala zauważyć pewne regularności, które trudno byłoby wykryć metodami czysto tekstowymi lub numerycznymi. Sekcje nagłówków często tworzą jasne, regularne pasma, natomiast dane wykonywalne przybierają formę obszarów o zróżnicowanej intensywności, odpowiadających różnym segmentom kodu. W przypadku plików zaszyfrowanych lub spakowanych pojawiają się obszary o wysokiej entropii, które wizualnie manifestują się jako ziarna o losowej strukturze. Takie różnice w układzie wizualnym mogą być skutecznie wykorzystywane przez modele głębokiego uczenia, które są zdolne do automatycznego wydobywania cech z obrazów bez konieczności ich ręcznego definiowania.

Koncepcja wizualizacji złośliwego oprogramowania znajduje zastosowanie zarówno w analizie statycznej, jak i dynamicznej. W analizie statycznej obrazy powstają bez uruchamiania pliku, jedynie na podstawie jego struktury binarnej. W analizie dynamicznej możliwe jest natomiast generowanie wizualnych reprezentacji przebiegu działania programu, na przykład poprzez tworzenie obrazów z ciągów wywołań API lub śladów wykonywania kodu. Choć ta druga forma jest bardziej złożona obliczeniowo, pozwala ona na uchwycenie cech behawioralnych, które mogą być niedostępne w analizie statycznej. Wizualizacja - jako sposób wykrywania złośliwego oprogramowania - wykorzystywana jest m.in. w pracach Nisa i wsp.[2], Anandhi i wsp.[3] czy Xiao i wsp.[4]

Typy wizualizacji plików binarnych

Najbardziej rozpowszechnioną metodą wizualizacji złośliwego oprogramowania jest obrazowanie w skali szarości. W tym podejściu każdy bajt pliku binarnego jest interpretowany jako wartość z zakresu od 0 do 255, odpowiadająca intensywności piksela w obrazie monochromatycznym. Dane te są następnie umieszczane w dwuwymiarowej macierzy, zwykle o szerokości będącej wielokrotnością 256, co pozwala na zachowanie proporcji i ułatwia analizę. W efekcie powstają charakterystyczne wzorce wizualne, w których różne sekcje pliku przybierają odmienne formy geometryczne. Tego rodzaju obrazy można bezpośrednio analizować przy użyciu konwolucyjnych sieci neuronowych (Convolutional Neural Network, CNN), które wykrywają w nich struktury odpowiadające cechom klasowym. Podejście to jest szeroko wykorzystywane m.in. w pracach badawczych Nisa i wsp.[5], Nataraj i wsp.[6] oraz Kalash i wsp.[7]

---

[2] M. Nisa, J.H. Shah, S. Kanwal, M. Raza, M.A. Khan, R. Damaševičius, T. Blažauskas, Hybrid Malware Classification Method Using Segmentation-Based Fractal Texture Analysis and Deep Convolution Neural Network Features, “Applied Sciences” 2020, no. 10(14), s. 4966, https://doi.org /10.3390/app10144966.

[3] V. Anandhi, P. Vinod, V.G. Menon, Malware visualization and detection using DenseNets, “Personal and Ubiquitous Computing” 2021, https://doi.org/10.1007/s00779-021-01581.

[4] G. Xiao, J. Li, Y. Chen, K. Li, MalFCS: An effective malware classification framework with automated feature extraction based on deep convolutional neural networks, “Journal of Parallel and Distributed Computing” 2020, no. 141, s. 49–58, https://doi.org/10.1016/j.jpdc.2020.03.012.

[5] M. Nisa, et.al., op.cit., s. 4966.

[6] L. Nataraj, el.al., op.cit., s. 1-7.

[7] M. Kalash, M. Rochan, N. Mohammed, N.D. Bruce, Y. Wang, M.D. Levine, Malware classification with deep convolutional neural networks. 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS) 2018, https://doi.org/10.1109/NTMS.2018.8328749.

Rys. 1 *Schemat przekształcania pliku binarnego do obrazu w skali szarości.*

Źródło: *opracowanie własne na podstawie L. Nataraj i inni*[8].

Zawartość pliku wykonywalnego jest odczytywana jako sekwencja bajtów, z których każdy reprezentuje 8-bitową liczbę całkowitą bez znaku w zakresie od 0 do 255. Wartości te są następnie rozmieszczane w postaci dwuwymiarowej macierzy, gdzie intensywność poszczególnych pikseli odpowiada równoważnym wartościom dziesiętnym bajtów, tworząc wizualną reprezentację struktury danych pliku. Jako przykład przedstawiono wizualizacje trzech próbek złośliwego oprogramowania należących do rodziny FakeRean[9].

Drugim istotnym podejściem jest wizualizacja kolorowa, znana jako reprezentacja RGB (Red, Green, Blue). W tym wariancie bajty pliku są grupowane po trzy i przypisywane odpowiednio do kanałów czerwonego, zielonego i niebieskiego. W efekcie uzyskuje się kolorowe obrazy, które mogą zawierać więcej informacji o relacjach pomiędzy bajtami, ponieważ każdy piksel odzwierciedla jednocześnie trzy wartości. Technikę tą zastosowali m.in. Ouahab i wsp.[10] Inne zastosowanie obrazów w formacie RGB zaproponowali Han i wsp.[11] wykorzystując sekwencje opcode'ów, które następnie są zamieniane przy pomocy funkcji hashującej na obrazy RGB. Natomiast, Zhao i wsp.[12] w pracy z 2023 roku proponują wygenerowanie obrazów w skali szarości przy pomocy trzech różnych technik: obrazy ASCII, obrazy szesnastkowe (heksadecymalne) oraz obrazy wygenerowane przy użyciu entropii. Każdy z nich jest traktowany jako osobny kanał RGB.

W badaniach przeprowadzonych przez zespół Anandhi i wsp.[13] porównują oni trzy rodzaje wizualizacji: wizualizację w odcieniach szarości, w reprezentacji RGB oraz w reprezentacji w postaci obrazów Markowa. W tym przypadku autorzy nadają każdemu kanałowi tę samą wartość pochodzącą z obrazu w skali szarości. Tak przygotowane obrazy są następnie klasyfikowane przy użyciu sieci DenseNet[14]. Dzięki zastosowaniu obrazów Markowa autorzy otrzymali prawie bezbłędną poprawność klasyfikacji oraz osiągnęli poprawę wydajności zarówno pod względem wykrywania, klasyfikacji, jak i czasu wykonania.

---

[8] L. Nataraj, et.al., op.cit., s. 1-7.

[9] https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Win32/FakeRean [dostęp: 21.12.2025].

[10] I.B.A. Ouahab, Y. Alluhaidan, L. Elaachak, M. Bouhorma, Malware Detection Using RGB Images and CNN Model Subclassing, [in:] A. A. Abd El-Latif, Y. Maleh, W. Mazurczyk, M. ELAffendi, M.I. Alkanhal (eds.), Advances in Cybersecurity, Cybercrimes, and Smart Emerging Technologies, Vol. 4, s. 3-13, Cham 2023, https://doi.org/10.1007/ 978-3-031-21101-0_1.

[11] K. Han, B. Kang, E.G. Im, Malware Analysis Using Visualized Image Matrices, "The Scientific World Journal" 2014, s. 1-15, https://doi.org/10.1155/2014/132713.

12 Z. Zhao, S. Yang, D. Zhao, A New Framework for Visual Classification of Multi-Channel Malware Based on Transfer Learning, "Applied Sciences" 2023, no. 13(4), s. 2484, https://doi.org/10.3390/app13042484.

13 V. Anandhi, el.al, op.cit.

14 Z. Zhao, et.al., op.cit., s. 2484.

Trzecim ważnym nurtem badań nad wizualizacją złośliwego oprogramowania jest obrazowanie oparte na entropii. Entropia, rozumiana jako miara nieuporządkowania informacji, odgrywa kluczową rolę w analizie plików spakowanych lub zaszyfrowanych. W metodach entropijnych plik binarny dzielony jest na równe fragmenty, dla których obliczana jest entropia Shannona. Uzyskane wartości są następnie odwzorowywane graficznie, najczęściej w postaci mapy ciepła lub wykresu gradientowego, w którym jasne obszary odpowiadają fragmentom o wysokiej entropii, a ciemne - fragmentom uporządkowanym, zawierającym kod lub dane jawne. Wizualizacja entropijna umożliwia szybkie wykrycie sekcji spakowanych lub zaszyfrowanych, co czyni ją przydatnym narzędziem w analizie złośliwego oprogramowania typu ransomware i oprogramowania wykorzystującego techniki zaciemniania kodu. To podeście zostało zaprezentowane m.in. u Xiao i wsp.[15], pozwalając osiągnąć wysoką dokładność, odporność na większość technik zaciemniania oraz wykazując się lepszą reprezentatywnością cech rodzin dzięki czemu uniknięcie wykrycia jest trudniejsze.

W literaturze przedmiotu pojawiają się także podejścia hybrydowe, w których łączy się wizualizację binarną z analizą semantyczną. Przykładowo, niektóre badania proponują tworzenie map cech, w których kolory odpowiadają różnym typom instrukcji asemblera, lub tworzenie obrazów z danych dynamicznych, takich jak ślady wywołań systemowych. Takie podejścia poszerzają możliwości klasyfikacji, łącząc zalety analizy wizualnej z informacją o zachowaniu programu. Przykładowo Li i wsp.[16] wykorzystali cechy zarówno z instrukcji asemblera, surowych danych binarych oraz wywołań API.

Modele uczenia maszynowego w analizie obrazów złośliwego oprogramowania

Zastosowanie wizualizacji plików binarnych otworzyło drogę do wykorzystania metod przetwarzania obrazów w dziedzinie cyberbezpieczeństwa. Dzięki temu badacze mogą korzystać z dojrzałych technologii opracowanych pierwotnie dla rozpoznawania wzorców wizualnych, takich jak klasyfikacja obrazów naturalnych, detekcja obiektów czy segmentacja semantyczna. W kontekście analizy złośliwego oprogramowania najczęściej stosowane są sieci konwolucyjne, które automatycznie uczą się hierarchicznych reprezentacji obrazu, począwszy od prostych krawędzi i tekstur, aż po złożone wzorce strukturalne. W porównaniu z klasycznymi metodami ekstrakcji cech, takimi jak histogramy wartości bajtów czy analiza n-gramów, modele konwolucyjne pozwalają na znacznie większą autonomię w uczeniu się reprezentacji.

Pierwsze badania nad wykorzystaniem konwolucyjnych sieci neuronowych w analizie złośliwego oprogramowania opartego na wizualizacji pojawiły się około roku 2016, kiedy to badacze zauważyli, że CNN są w stanie wykrywać unikalne wzorce w obrazach pochodzących z plików wykonywalnych. Kalash i wsp.[17] przedstawili model oparty na architekturze CNN, który klasyfikował próbki złośliwego oprogramowania w skali odcieni szarości. Uzyskane wyniki wykazały, że nawet niewielkie sieci konwolucyjne mogą osiągać

15 G. Xiao, el.al, op.cit., s. 49.
16 S. Li, J. Wang, Y. Song, S. Wang, Tri-channel visualised malicious code classification based on improved ResNet, “Applied Intelligence” 2024, no. 54(23), s. 12453–12475, https://doi.org/10.1007/s10489-024-05707-4.
17 M. Kalash, et.al., op.cit.

dokładność przekraczającą 98% w rozróżnianiu głównych rodzin złośliwego oprogramowania. W dalszych pracach wprowadzono bardziej złożone architektury, takie jak ResNet[18], VGG3[19], VGG16 (Rezende i wsp.[20] wykorzystali cechy z warstwy bottleneck) czy EfficientNet[21], DenseNet, AlexNet i Inception-V3[22] które dzięki głębszym warstwom konwolucyjnym i zastosowaniu mechanizmów normalizacji pozwalają na uzyskanie jeszcze lepszych wyników przy ograniczonym ryzyku przeuczenia.

Oprócz klasycznych sieci konwolucyjnych w analizie wizualizacji plików binarnych wykorzystuje się także autoenkodery[23], czyli modele uczące się rekonstrukcji danych wejściowych. Autoenkoder składa się z dwóch części: kodera, który przekształca obraz w przestrzeń reprezentacji latentnej, oraz dekodera, który próbuje odtworzyć pierwotny obraz na podstawie tej reprezentacji. Porównanie obrazu wejściowego z obrazem zrekonstruowanym pozwala wykryć odchylenia, które mogą wskazywać na anomalie. W praktyce oznacza to, że autoenkoder uczony na obrazach pochodzących z oprogramowania niegroźnego będzie miał trudność z poprawną rekonstrukcją obrazu przedstawiającego złośliwe oprogramowanie. Taka różnica może służyć jako wskaźnik podejrzanego zachowania. Modele tego typu znalazły zastosowanie zwłaszcza w detekcji wcześniej nieznanych rodzin złośliwego oprogramowania, w sytuacjach, gdy etykietowanie danych jest ograniczone lub kosztowne.

Kolejnym kierunkiem rozwoju metod wizualnej analizy złośliwego oprogramowania są transformery wizualne (Vision Transformers, ViT), które przenoszą rozwiązania opracowane dla przetwarzania języka naturalnego na grunt analizy obrazów. Model ViT dzieli obraz na małe fragmenty (tzw. patches), które traktowane są jako sekwencje wektorów wejściowych. Mechanizm samouwagi (self-attention) pozwala modelowi koncentrować się na fragmentach obrazu najbardziej istotnych dla klasyfikacji. W badaniach[24] wykazano, że transformery wizualne wykazują większą odporność na zmienność danych niż klasyczne sieci konwolucyjne, co jest szczególnie istotne w przypadku złośliwego oprogramowania, które często ewoluuje poprzez niewielkie modyfikacje strukturalne.

W analizie obrazów pochodzących z plików binarnych wykorzystywane są również metody klasteryzacji nienadzorowanej, które pozwalają grupować próbki

---

[18] S. Li, el.al., op.cit., s. 12453.
[19] Z. Zhao, et.al., op.cit., s. 2484.
[20] E. Rezende, G. Ruppert, T. Carvalho, A. Theophilo, F. Ramos, P. Geus, Malicious Software Classification Using VGG16 Deep Neural Network's Bottleneck Features, [in:] Latifi, S. (eds.) Information Technology - New Generations. Advances in Intelligent Systems and Computing, vol. 738, Cham 2018, https://doi.org/10.1007/978-3-319-77028-4_9.
[21] V. Ravi, R. Chaganti, EfficientNet deep learning meta-classifier approach for image-based android malware detection, "Multimedia Tools and Applications" 2023, vol. 82(16), s. 24891–24917, https://doi.org/10.1007/s11042-022-14236-6.
[22] M. Nisa, el.al., op.cit., s. 4966.
[23] J. Lee, J.A. Lee, Classification System for Visualized Malware Based on Multiple Autoencoder Models, "IEEE Access" 2021, vol. 9, s. 144786-144795, https://doi.org/10.1109/access.2021.3122083.
[24] S. Seneviratne, R. Shariffdeen, S. Rasnayaka, N. Kasthuriarachchi, Self-Supervised Vision Transformers for Malware Detection, "IEEE Access" 2022, vol. 10, s. 103121-103135, https://doi.org/10.1109/access.2022.3206445; M. Ashawa, N. Owoh, S. Hosseinzadeh, J. Osamor, Enhanced Image-Based Malware Classification Using Transformer-Based Convolutional Neural Networks (CNNs) "Electronics" 2024, vol. 13(20), s. 4081, https://doi.org/10.3390/electronics13204081.

o podobnych cechach wizualnych bez potrzeby wcześniejszego etykietowania. Takie podejście umożliwia identyfikację nowych, nieznanych rodzin złośliwego oprogramowania poprzez obserwację, które próbki tworzą nowe skupiska w przestrzeni cech. W tym kontekście często wykorzystuje się techniki redukcji wymiarowości, takie jak analiza składowych głównych (Principal Component Analysis, PCA) lub t-SNE (t-distributed Stochastic Neighbor Embedding), które ułatwiają wizualizację struktury danych i wnioskowanie o podobieństwie pomiędzy próbkami.

Zastosowania praktyczne i wyniki badań

Metody wizualizacji plików binarnych znalazły szerokie zastosowanie zarówno w badaniach akademickich, jak i w praktyce. Najczęstszym celem jest klasyfikacja rodzin złośliwego oprogramowania, w której model uczy się rozpoznawać wzorce wizualne charakterystyczne dla danej grupy próbek. W literaturze można znaleźć przykłady modeli osiągających bardzo wysokie wyniki dokładności w klasyfikacji wieloklasowej, często przekraczające 95%. W badaniu Nataraj i wsp.[25], w którym zastosowano proste obrazy w skali szarości i klasyfikator k-NN, uzyskano skuteczność rzędu 97%. W kolejnych latach wprowadzenie sieci konwolucyjnych pozwoliło przekroczyć granicę 99% dokładności w zadaniach obejmujących kilka najczęstszych rodzin złośliwego oprogramowania.

Oprócz klasyfikacji rodzin, wizualizacja złośliwego oprogramowania wykorzystywana jest także w analizie podobieństw i genealogii kodu. Badacze zauważyli, że obrazy pochodzące z plików o wspólnym pochodzeniu wykazują silne podobieństwa teksturalne i strukturalne, co umożliwia odtwarzanie relacji pomiędzy próbkami. Takie analizy są szczególnie użyteczne w kontekście badań nad ewolucją złośliwego oprogramowania oraz identyfikacji źródeł kampanii cyberataków.

Istotnym zastosowaniem wizualizacji jest również wykrywanie obfuskacji. Obszary o wysokiej entropii, które w obrazie manifestują się jako chaotyczne fragmenty o losowej strukturze, mogą wskazywać na sekcje spakowane lub zaszyfrowane. Analiza takich fragmentów pozwala na szybkie wykrycie technik zaciemniania kodu i automatyczne oznaczanie próbek wymagających głębszej analizy. Wizualne wskaźniki obfuskacji znajdują zastosowanie w narzędziach wspomagających analizę sądową, gdzie analityk może szybko ocenić, które sekcje pliku wymagają dekompresji lub deszyfrowania.

W badaniach praktycznych zwraca się także uwagę na możliwość integracji metod wizualnych z innymi technikami analizy, takimi jak analiza zachowania czy analiza sieci wywołań API. Łączenie danych wizualnych z informacjami semantycznymi tworzy tzw. modele multimodalne, które analizują próbki złośliwego oprogramowania na wielu poziomach reprezentacji. Tego rodzaju podejścia są uważane za najbardziej obiecujące, ponieważ łączą odporność metod wizualnych na obfuskację z precyzją metod semantycznych w rozpoznawaniu funkcjonalności kodu.

Ograniczenia i wyzwania podejścia wizualnego

[25] L. Nataraj, et.al., op.cit. s. 7.

Pomimo wielu zalet, podejście oparte na wizualizacji złośliwego oprogramowania nie jest pozbawione ograniczeń. Najczęściej wskazywanym problemem jest utrata informacji semantycznych podczas przekształcania pliku binarnego w obraz. Choć struktura bajtowa może odzwierciedlać pewne wzorce, nie zachowuje ona kontekstu wykonywania instrukcji ani relacji pomiędzy sekcjami kodu. W rezultacie dwa pliki o identycznym zachowaniu, ale różnej strukturze binarnej, mogą tworzyć zupełnie różne reprezentacje wizualne, co utrudnia generalizację modeli.

Kolejnym ograniczeniem jest zależność od parametrów technicznych wizualizacji, takich jak szerokość obrazu, sposób grupowania bajtów czy metoda normalizacji. W literaturze wskazuje się, że zmiana tych parametrów może znacząco wpłynąć na wyniki klasyfikacji. Brak jednolitych standardów utrudnia porównywanie rezultatów pomiędzy badaniami i tworzenie wspólnych benchmarków.

Istotnym wyzwaniem jest również podatność modeli wizualnych na dryf koncepcji (concept drift), czyli stopniową zmianę charakterystyki danych w czasie. W miarę jak pojawiają się nowe rodziny złośliwego oprogramowania, wcześniej wytrenowane modele mogą tracić skuteczność, ponieważ wzorce wizualne ulegają zmianie. Problem ten dotyczy nie tylko wizualizacji, lecz także wszystkich modeli uczenia maszynowego w cyberbezpieczeństwie. W przypadku obrazów plików binarnych jest jednak szczególnie widoczny, ponieważ zmiana choćby niewielkiej części kodu może prowadzić do całkowitej zmiany struktury obrazu.

Kolejną barierą dla szerokiego zastosowania metod wizualnych jest brak powszechnie dostępnych, aktualnych zbiorów danych w formacie obrazowym. Większość istniejących zestawów danych, takich jak EMBER[26], DREBIN[27] czy MalNet[28], zawiera dane w postaci wektorów cech lub metadanych, natomiast zbiory obrazów pochodzących z plików binarnych są wciąż tworzone przez poszczególne zespoły badawcze na własne potrzeby. Brak standaryzacji w tym zakresie utrudnia reprodukcję wyników oraz porównywanie skuteczności różnych metod.

Kierunki dalszych badań i podsumowanie

Kierunki rozwoju metod wizualizacji złośliwego oprogramowania koncentrują się obecnie wokół trzech głównych obszarów. Pierwszym z nich jest rozwój modeli multimodalnych, łączących dane wizualne z informacjami pochodzącymi z analizy zachowania i struktury kodu. Drugim jest wprowadzenie mechanizmów uczenia ciągłego (Continuous/Incremental Learning), które umożliwiają modelom szybką adaptację do nowych rodzin złośliwego

[26] H. S. Anderson, P. Roth, EMBER: An Open Dataset for Training Static PE Malware Machine Learning Models (Version 2), arXiv. 2018, https://doi.org/10.48550/arXiv.1804.04637; R. J. Joyce, G. Miller, P. Roth, R. Zak, E. Zaresky-Williams, H. Anderson, E. Raff, J. Holt, MBER2024 – A Benchmark Dataset for Holistic Evaluation of Malware Classifiers. Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, 2025, s. 5516–5526. https://doi.org/10.1145/3711896.3737431.

[27] D. Arp, M. Spreitzenbarth, M. Hübner, H. Gascon, K. Rieck, Drebin: Effective and Explainable Detection of Android Malware in Your Pocket, Proceedings 2014 Network and Distributed System Security Symposium, Network and Distributed System Security Symposium, San Diego, CA. https://doi.org/10.14722/ndss.2014.23247.

[28] S. Freitas, R. Duggal, D.H. Chau, MalNet: A Large-Scale Image Database of Malicious Software (No. arXiv:2102.01072). arXiv., https://doi.org/10.48550/arXiv.2102.01072.

oprogramowania poprzez dostrajanie (fine-tuning) lub przyrostowe aktualizowanie już istniejących wag, eliminując konieczność kosztownego i czasochłonnego ponownego trenowania na całym historycznym zbiorze danych. Trzecim jest rozwój metod interpretowalności modeli głębokiego uczenia, które pozwalają zrozumieć, jakie fragmenty obrazu wpływają na decyzję klasyfikacyjną.

W perspektywie długoterminowej wizualizacja plików binarnych ma szansę stać się integralnym elementem systemów bezpieczeństwa informatycznego. Jej siła tkwi w zdolności do ujawniania ukrytych struktur w danych, które są trudne do uchwycenia klasycznymi metodami analizy. Połączenie jej z innymi źródłami informacji, takimi jak logi systemowe, sygnatury sieciowe czy analiza dynamiczna, może doprowadzić do powstania zintegrowanych systemów detekcji, zdolnych do reagowania na nowe zagrożenia w sposób bardziej elastyczny i odporny.

Wizualizacja złośliwego oprogramowania stanowi przykład interdyscyplinarnej integracji metod informatyki, analizy danych i przetwarzania obrazu, umożliwiającej bardziej złożone i interpretowalne podejście do detekcji oraz klasyfikacji zagrożeń. Choć wciąż wymaga standaryzacji i dalszych badań, już dziś dostarcza narzędzi, które znacząco poszerzają możliwości analizy i klasyfikacji złośliwego oprogramowania. Ostatecznym celem tych prac jest stworzenie systemów, które nie tylko rozpoznają znane zagrożenia, ale potrafią samodzielnie dostosowywać się do nowych form ataków, a wizualizacja plików binarnych może odegrać w tym procesie istotną rolę.

Bibliografia

Anandhi V., Vinod P., Menon V.G., Malware visualization and detection using DenseNets, “Personal and Ubiquitous Computing” 2021, https://doi.org/10.1007/s00779-021-01581-w.

Anderson H.S., Roth P., EMBER: An Open Dataset for Training Static PE Malware Machine Learning Models (Version 2), arXiv. 2018, https://doi.org/10.48550/arXiv.1804.04637.

Arp D., Spreitzenbarth M., Hübner M., Gascon H., Rieck K., Drebin: Effective and Explainable Detection of Android Malware in Your Pocket, Proceedings 2014 Network and Distributed System Security Symposium, Network and Distributed System Security Symposium, San Diego, CA. https://doi.org/10.14722/ndss.2014.23247.

Ashawa M., Owoh N., Hosseinzadeh S., Osamor J., Enhanced Image-Based Malware Classification Using Transformer-Based Convolutional Neural Networks (CNNs) “Electronics” 2024, vo. 13(20), https://doi.org/10.3390/ electronics13204081.

Freitas S., Duggal R., Chau D.H., MalNet: A Large-Scale Image Database of Malicious Software (No. arXiv:2102.01072), arXiv., https://doi.org/10.48550/arXiv.2102.01072.

Han K., Kang B., Im E.G., Malware Analysis Using Visualized Image Matrices, “The Scientific World Journal” 2014, https://doi.org/10.1155/2014/132713.

Joyce R.J., Miller G., Roth P., Zak R., Zaresky-Williams E., Anderson H., Raff E., Holt J., MBER2024 - A Benchmark Dataset for Holistic Evaluation of Malware Classifiers. Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, 2025, https://doi.org/10.1145/3711896.3737431.

Kalash M., Rochan M., Mohammed N., Bruce N.D, Wang Y., Levine M.D., Malware classification with deep convolutional neural networks. 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS) 2018, https://doi.org/10.1109/NTMS.2018.8328749.

Lee J., Lee J.A., Classification System for Visualized Malware Based on Multiple Autoencoder Models, “IEEE Access” 2021, vol. 9, https://doi.org/10.1109/ access.2021. 3122083.

Li S., Wang J., Song Y., Wang S., Tri-channel visualised malicious code classification based on improved ResNet, “Applied Intelligence” 2024, no. 54(23), https://doi. org/10.1007/s10489-024-05707-4.

Nataraj L., Karthikeyan S., Jacob G., Manjunath B.S., Malware images: visualization and automatic classification, Proceedings of the 8th International Symposium on Visualization for Cyber Security (VizSec’11) 2011, no.4, https://doi.org/10.1145/ 2016904.2016908.

Nisa M., Shah J.H., Kanwal S., Raza M., Khan M.A., Damaševičius R., Blažauskas T., Hybrid Malware Classification Method Using Segmentation-Based Fractal Texture Analysis and Deep Convolution Neural Network Features, “Applied Sciences” 2020, 10(14), https://doi.org /10.3390/app10144966.

Ouahab I.B.A., Alluhaidan Y., Elaachak L., Bouhorma M., Malware Detection Using RGB Images and CNN Model Subclassing, [in:] A. A. Abd El-Latif, Y. Maleh, W. Mazurczyk, M. ELAffendi, M.I. Alkanhal (eds.), Advances in Cybersecurity, Cybercrimes, and Smart Emerging Technologies, Vol. 4, Springer International Publishing 2023, https://doi.org/10.1007/ 978-3-031-21101-0_1.

Ravi V., Chaganti R., EfficientNet deep learning meta-classifier approach for image-based android malware detection, “Multimedia Tools and Applications” 2023, vol. 82(16), https://doi.org/10.1007/ s11042-022-14236-6.

Rezende E., Ruppert G., Carvalho T., Theophilo A., Ramos F., Geus P., Malicious Software Classification Using VGG16 Deep Neural Network’s Bottleneck Features, [in:] Latifi, S. (eds.) Information Technology - New Generations. Advances in Intelligent Systems and Computing, vol 738, Springer Cham 2018, https://doi.org/10.1007/978-3-319-77028-4_9.

Seneviratne S., Shariffdeen R., Rasnayaka S., Kasthuriarachchi N., Self-Supervised Vision Transformers for Malware Detection, “IEEE Access” 2022, vol. 10, https://doi.org/10.1109/access.2022.3206445.

Xiao G., Li J., Chen Y., Li K., MalFCS: An effective malware classification framework with automated feature extraction based on deep convolutional neural networks, “Journal of Parallel and Distributed Computing” 2020, no. 141, https://doi.org/ 10.1016/j.jpdc.2020.03.012.

Zhao Z., Yang S., Zhao D., A New Framework for Visual Classification of Multi-Channel Malware Based on Transfer Learning, “Applied Sciences” 2023, no. 13(4), https://doi.org/10.3390/ app13042484.