https://doi.org/10.7840/kics.2025.50.11.1739

딥러닝 기반 이미지 분석을 활용한 악성 네트워크 트래픽 탐지 동향

박 영 주◆, 이 선 우°

A Survey on Deep Learning-Based Image Approaches for Malicious Network Traffic Detection

Young-joo Park◆, Sun Woo Lee°

요 약

사이버 공격이 점점 정교해짐에 따라 네트워크 트래픽을 기반으로 악성 행위를 탐지하는 기술의 중요성이 증가하고 있다. 최근에는 네트워크 트래픽을 Grayscale 또는 RGB 이미지로 변환한 후, 딥러닝 모델을 활용해 이를 분류하는 이미지 기반 탐지 기법이 주목받고 있다. 본 논문은 이러한 기법들을 모델 구조(CNN, CNN-LSTM, GAN, GNN 등), 이미지 처리 방식, IoT 환경 등 대상 조건, 실험 설정에 따라 체계적으로 분류·분석하였다. 또한, 학습되지 않은 공격을 탐지하기 위한 OSR(Open-set Recognition) 기반 기법도 함께 고찰하였다. 본 연구는 다양한 전처리 전략과 이미지화 방식, OSR 적용 사례를 종합적으로 정리하고, IoT 환경에 적합한 경량화, 실시간 처리, 프로토콜 독립성 확보, 고도화된 OSR 기법 등 향후 연구 방향을 제시한다.

키워드 : 악성코드 트래픽 탐지, 악성 트래픽 탐지, 이미지 기반 분류
Key Words : Malware Traffic Detection, Malicious Traffic Detection, Image-Based Classification

ABSTRACT

As cyberattacks grow more sophisticated, detecting malicious behavior through network traffic has become increasingly critical. Recently, image-based detection approaches that convert raw network traffic into grayscale or RGB images for deep learning classification have gained attention. This paper provides a structured survey on such methods, categorizing existing research by model architecture (e.g., CNN, CNN-LSTM, GAN, GNN), image processing techniques, target environments such as IoT, and evaluation settings. We also examine open-set recognition (OSR) approaches aimed at identifying previously unseen attacks. By analyzing preprocessing strategies, image encoding methods, and OSR applications, this study outlines key limitations of current research and discusses future directions, including lightweight modeling for IoT, real-time performance, protocol independence, and enhanced OSR techniques.

※ 이 성과(논문)는 정부(교육부)의 지원을 받아 수행된 연구임 (2024년 부처 협업형 인재양성사업[정보보안 분야], No. 2024개인정보보호-002)
◆ First Author : Seoul Women's University of Department of Information Security, parkzer0joo@gmail.com, 학생회원
° Corresponding Author : Seoul Women's University of Department of Information Security, sun.lee@swu.ac.kr, 정회원
논문번호 : 202504-088-C-RN, Received April 15, 2025; Revised June 26, 2025; Accepted June 27, 2025

Ⅰ. 서 론

인터넷과 디지털 서비스의 확산으로 사이버 공격은 빈도와 정교함이 증가하고 있으며, 이는 산업 전반과 일상에 심각한 보안 위협을 초래하고 있다[1]. 특히, ChatGPT와 같은 생성형 인공지능 기술의 발전으로 일반 사용자도 손쉽게 공격을 시도할 수 있는 환경이 조성되면서 위협 수준은 더욱 높아지고 있다. 이에 따라 악성 행위를 조기에 탐지하고 대응하기 위한 네트워크 기반 보안 기술의 중요성이 부각되고 있다.

기존의 트래픽 기반 탐지 기법은 포트 기반 탐지, 심층 패킷 검사, 통계 기반 탐지로 구분된다[2]. 이 중 시그니처 기반 방식인 포트 기반 및 패킷 검사는 알려진 공격에는 효과적이나, 암호화된 트래픽 분석과 시그니처 관리 측면에서 한계를 지닌다. 통계 기반 탐지 기법은 암호화된 트래픽에도 적용 가능하지만, 공격자의 패턴 조작에 취약하며 새로운 공격 유형에 대한 유연성이 부족하다. 이러한 한계를 극복하기 위해 최근에는 딥러닝 기반의 자동화된 탐지 기법이 주목받고 있다. 특히, 네트워크 트래픽을 이미지로 변환하고 CNN 등 딥러닝 모델로 학습하는 방식은 수동 특징 설계 없이도 높은 탐지 성능을 보여 활발히 연구되고 있다[3]. 본 논문은 이미지 기반 악성 트래픽 탐지 기법을 데이터셋, 이미지 처리 방식, 모델 구조, 탐지 성능 등의 측면에서 체계적으로 비교·분석한다. 특히 기존의 이진 분류 중심에서 다중 분류 및 Open-set Recognition(OSR) 기반 탐지로 확장되는 최근 흐름에 주목하며, 각 기법의 한계와 실제 적용 가능성을 고찰하고 향후 연구 방향을 제시한다.

본 논문의 구성은 다음과 같다. 본 논문은 제2장에서 주요 데이터셋을, 제3장에서 트래픽의 이미지화 방식을, 제4장에서 딥러닝 기반 탐지 구조와 성능을 다룬다. 제5장에서는 기존 연구의 한계와 과제를, 제6장에서 결론을 제시한다.

Ⅱ. 악성 트래픽 데이터 셋

이미지 기반 트래픽 분석에는 일반적으로 pcap 형식의 원본 네트워크 트래픽이 사용된다. 일부 연구는 실험 환경에서 트래픽을 직접 수집하기도 하지만, 대부분은 연구 목적에 부합하는 공개 데이터셋을 활용한다. 이러한 데이터셋은 생성 방식과 탐지 대상에 따라 악성코드 트래픽, 악성 트래픽, 암호화 트래픽 등으로 구분된다. 예를 들어, 악성코드 탐지를 목표로 하는 경우에는 악성코드 실행 중 발생한 트래픽을 포함한 '악성코드 트래픽'이 사용되며, 특정 공격 행위를 탐지하려는 경우에는 다양한 공격 시나리오가 반영된 '악성 트래픽' 데이터셋이 적합하다. 암호화 환경에서의 분석을 목표로 하는 연구에는 '암호화 트래픽' 데이터셋이 활용된다. 이처럼 데이터셋의 유형은 연구 목적과 탐지 대상에 따라 달라지며, 본 장에서는 이미지 기반 탐지 연구에서 널리 사용되는 대표적인 데이터셋들을 세 가지 범주로 나누어 소개한다.

2.1 악성코드 트래픽 데이터셋

ISOT HTTP Botnet Dataset[4]은 HTTP 기반 C&C 통신을 수행하는 Zeus 봇넷 트래픽과 정상 웹 트래픽을 포함해, DNS 기반 봇넷 탐지 연구에 적합하다. Malware Capture Facility Project (MCFP)[5]는 Zeus, Sality, Conficker 등 다양한 악성코드를 실제 실행해 생성된 트래픽을 수집한 데이터셋으로, C&C 통신, 스팸 발송, 정보 탈취 등 다양한 공격 시나리오를 포함하며, 각 세션은 악성 트래픽 중심으로 구성된다.

표 1. 데이터셋 비교
Table 1. Comparison of datasets

Dataset	Data Type	Collection Environment	Collection Method	Application
ISOT HTTP Botnet Dataset[4]	DNS traffic from HTTP-based botnets and benign application	Windows XP/7 VMs	Monitored DNS requests to custom DNS and C&C servers	DNS-based botnet C&C detection
MCFP[5]	Network traffic from malware	Virtual Machines	Captured traffic from malware using tcpdump and other analysis tools	Behavioral analysis of malware communication patterns
CICAndMal'17[6]	Android traffic from malicious apps (42 families in 4 categories)	Real Android smartphones (Nexus 4/5)	Captured during app lifecycle using CICFlowMeter, later extended to include permissions, intents, and API call sequences.	Android malware detection, category classification using multi-source features

Dataset	Data Type	Collection Environment	Collection Method	Application
CICInves AndMal'19[7]	Android traffic from malicious apps (42 families in 4 categories)	Real Android smartphones (Nexus 4/5)	Captured API call sequences and aggregated network flow features	Android malware classification using network and behavioral API features
CICMalDorid 2020[8]	Android traffic from malicious apps (5 categories)	Android VMs (CopperDroid sandbox)	Captured system calls, binder calls, composite behaviors, and network traffic	Behavioral profiling and malware category analysis based on dynamic behaviors
NSL-KDD[9]	TCP network traffic from attack connections	Simulated military LAN (U.S. Air Force LAN)	Processed from raw TCP dump data collected	Detection and Calssification of malicious TCP flows
ISCXIDS2012[10]	TCP traffic from multi-stage attacks across multiple protocols	Windows XP SP1,2,3/7 VMs	Generated normal traffic using user behavior profiles and executed multi-stage attack scenarios	Multi-stage attack detection
CTU-13[11]	Network traffic from botnet malware	Windows XP SP2 VM	Captured botnet traffic by executing malware in bridged VMs and labeling NetFlows	Botnet traffic classification by attack type
UNSW-NB15[12]	Network traffic from CVE-based synthetic attacks	Virtual servers and routers with IXIA PerfectStorm	Generated normal and attack traffic using IXIA; captured with tcpdump and processed with Argus and Bro-IDS	Detection multi-protocol attack scenarios
CICIDS'17[13]	Network traffic from 7 attack types	Real network with Windows, Linux, macOS	Captured full packet data and labeled flows using CICFlowMeter	Intrusion detection of network attacks using flow-based data
CICIDS'18[13]	Realistic enterprise network traffic and profile-based benign behavior	AWS-based network using Windows 8.1/10, Ubuntu	Collected PCAP and system logs and extracted flow features using CICFlowMeter-V3	Anomaly detection in a multi-subnet enterprise setting
CICDDoS'19[14]	Network traffic from DDoS attacks	Ubuntu server, Windows 7~10, and Fortinet firewall	Captured PCAP in LAN testbed and extracted flow features using CICFlowMeter-V3	Detection DDoS attack
CICIoT'22[15]	Network traffic from IoT devices under Flood and RTSP Brute Force attacks	Real IoT lab with 60 devices (attacks on selected devices)	Captured attack traffic using Wireshark, dumpcap, and protocol-specific sniffers during controlled experiments	Detection and analysis of protocol-specific attacks in IoT
CICIoT'23[16]	Network traffic form IoT devices under 33 types of large-scale attacks	Real IoT lab with 105 devices	Captured traffic using Wireshark and network TAP, and extracted features with DPKT	Intrusion detection and classification of large-scale IoT attacks
ISCX VPN-nonVPN[17]	Labeled encrypted and VPN traffic across 7 application categories	Lab environment with OpenVPN	Collected traffic with tcpdump and labeled using ISCXFlowMeter	Flow-based classification of VPN and encrypted application traffic

CICAndMal2017[6]은 안드로이드 악성코드 탐지를 목적으로 제작된 데이터셋으로, 총 42개 악성코드 패밀리를 Adware, Ransomware, Scareware, SMS의 4가지 유형으로 분류하였다. 실제 스마트폰에 악성 앱을 설치해 수집한 트래픽을 기반으로 하며, 이후 연구에서는 시스템 로그, API 호출, CPU 사용량 등 정적 및 동적 특성도 포함되었다. CICInvesAndMal2019[7]는 동일한 악성코드 패밀리를 포함하되, 네트워크 플로우와 API 호출 시퀀스를 중심으로 특징을 추출한 데이터셋이다. 마지막으로, CICMalDroid2020[8]은 Adware, Banking, SMS, Riskware, Benign의 5개 유형을 포함하며, VMI 기반의 CopperDroid 도구를 통해 앱 실행 중 발생하는 시스템 호출과 네트워크 트래픽을 수집한 정적·동적 정보 기반의 데이터셋이다.

2.2 악성 트래픽 데이터셋

NSL-KDD[9]는 KDD Cup 1999 데이터셋의 중복 및 불균형 문제를 개선한 트래픽 데이터셋으로, DoS, R2L, U2R, Probe의 4가지 공격 유형과 하나의 정상 클래스를 포함한다. ISCXIDS2012[10]는 가상 네트워크 환경에서 7일간 정상 트래픽과 7가지 공격 유형(Brute Force SSH, DDoS, DoS, HTTP DoS, Infiltrating, Botnet, SQL Injection)을 수집하였으며, 현실적인 다단계 공격 시나리오를 반영한다. CTU-13[11]은 봇넷 탐지용 대표 데이터셋으로, 13개 시나리오 기반의 트래픽을 제공하며, DDoS, 스팸, 포트 스캐닝 등의 공격 유형을 포함한다. UNSW-NB15[12]는 IXIA PerfectStorm 도구로 생성된 트래픽을 기반으로 하며, Fuzzers, Analysis, Backdoor 등 CVE 기반 9가지 공격 유형을 포함한다. CICIDS2017[13]은 CICFlowMeter 도구를 통해 80개의 트래픽 특성을 추출하고, Pcap 및 CSV 형식으로 제공된다. Brute Force, Heartbleed, Botnet, DoS, DDoS, Web Attack, Infiltration 등 7가지 공격 유형이 포함된다. CSE-CIC-IDS2018[13]은 CICIDS2017의 확장판으로, AWS 기반 가상환경에서 실제 기업 네트워크를 시뮬레이션하여 수집되었다. Brute Force, Heartbleed, Botnet, DoS, DDoS, Web Attack, Infiltration 등 7가지 공격이 포함되며, 사용자 프로파일 기반 트래픽을 반영한다. CICFlowMeter-V3를 활용하여 84개의 플로우 기반 특성을 추출하고, Pcap 및 CSV 형식으로 제공된다. CIC-DDoS2019[14]는 DDoS 탐지에 특화된 데이터셋으로, TCP/UDP, HTTP Flooding, SYN Flooding, DNS, MSSQL, LDAP 등 다양한 서비스 기반 공격 유형을 포함한다. CICIoT2022[15]는 실제 IoT 기기를 대상으로 Flood 및 RTSP Brute Force 공격을 수행하여 트래픽을 수집한 데이터셋이다. 마지막으로, CICIoT2023[16]은 105개 IoT 기기를 활용한 대규모 실험 환경에서 33가지 공격을 수행하고, 이를 DDoS, DoS, Reconnaissance, Web 기반 공격, Brute Force, Spoofing, Mirai의 7개 범주로 분류하였다.

2.3 암호화 트래픽 데이터셋

ISCX VPN-nonVPN[17]은 암호화된 트래픽 분석 및 VPN 식별 연구를 위해 구축된 데이터셋으로, YouTube, Facebook, Skype, Gmail, Netflix 등 다양한 애플리케이션의 트래픽을 VPN 적용 여부에 따라 수집하였다. OpenVPN, L2TP, PPTP 등 여러 VPN 프로토콜이 포함된다.

Ⅲ. 트래픽 이미지화 기법

본 장에서는 네트워크 트래픽을 이미지 형태로 변환하는 기법을 설명한다. 일반적으로 네트워크 트래픽은 pcap 형식으로 수집되며, 이미지로 변환하기 전 여러 단계의 전처리가 필요하다. 트래픽 이미지화 방식은 크게 Grayscale 이미지화와 RGB 이미지화로 구분되며, 그림 1은 전체 과정을 시각화한 것이다.

3.1 네트워크 트래픽 전처리

트래픽 이미지화를 위해서는 수집된 pcap 데이터를 분할, 정제, 정규화하는 전처리 과정이 필요하다. 분할은 분석 단위를 정의하는 과정으로, 시퀀스 기준(Flow, Session) 또는 계층 기준(전체 계층, L7 계층)에 따라 나뉜다. 예를 들어, Flow는 동일한 5-tuple(출발지/목적지 IP·Port, 프로토콜)을 갖는 단방향 패킷의 집합이며,

그림 1. 네트워크 트래픽의 이미지 변환 과정
Fig. 1. Flowchart of network traffic to image conversion.

Session은 양방향 흐름을 포함한다. 이를 조합하면 (1) Flow-전체 계층, (2) Flow-L7, (3) Session-전체 계층, (4) Session-L7의 네 가지 유형으로 나눌 수 있다. 정제는 중복 데이터 제거 및 IP/MAC 주소의 익명화 등을 통해 학습의 편향을 줄이는 과정이다. 단, 동일 네트워크 내에서는 주소가 분석에 큰 영향을 미치지 않아 익명화를 생략하기도 한다. 정규화는 이미지 크기를 통일하는 과정으로, 부족한 바이트는 0x00으로 패딩된다. 이 크기는 모델 입력 요건에 따라 결정된다. 예를 들어, [20]은 Session-전체 계층 구성이 가장 높은 성능을 보였으며, 이후 다수의 연구[22,23,26-39]에서도 Flow 또는 Session 단위를 기준으로 앞부분 n바이트를 사용하고 나머지는 패딩하는 방식을 채택하였다. 한편, [32]는 4개의 연속된 패킷 중 앞 50바이트는 헤더, 뒤 50바이트는 페이로드로 구분하고 256바이트로 트리밍하는 구조를 제안하였다. 이는 트래픽의 구조를 유지하면서도 입력 형식을 만족시키는 접근이다.

3.2 Grayscale 이미지화

Grayscale 방식은 트래픽의 바이너리 데이터를 단일 채널의 픽셀로 매핑하는 방식이다. 대표적으로 byteplot 기법[18]은 각 바이트(0x00~0xFF)를 픽셀 밝기 값(0~255)으로 변환하며, 0은 검정, 255는 흰색으로 나타난다. 이는 원래 악성 코드 탐지에 활용되었으나[19], 이후 네트워크 트래픽에도 적용되며 이미지 기반 탐지 연구의 기반을 마련하였다[20].

3.3 RGB 이미지화

RGB 방식은 트래픽의 정보를 R, G, B 세 채널로 나누어 시각적으로 더 풍부하게 표현한다. 예를 들어, Binvis 도구[21,34]는 바이트 값을 ASCII 코드로 해석한 후, 범주별로 색상을 할당한다. 0x00은 검정, 0xFF는 흰색, 확장 ASCII는 빨강, 제어 문자는 초록, 일반 텍스트와 숫자는 파랑으로 매핑된다. 또 다른 접근은 트래픽을 헤더, 송신자 페이로드, 수신자 페이로드로 나누고, 각 요소를 마르코프 전이 행렬로 변환한 후 R, G, B 채널에 매핑하는 방식이다[25,32].

3.4 이미지화 방식 비교

Grayscale과 RGB 방식 모두 트래픽을 이미지로 변환해 딥러닝 모델에 입력할 수 있으나, 표현력, 연산량, 적용 대상에서 차이가 있다. Grayscale 방식은 단일 채널의 밝기 값만을 사용해 바이너리 패턴을 직관적으로 시각화하며, 계산 비용과 메모리 사용이 낮다. 그러나 트래픽의 구조적·행동적 특징을 표현하는 데는 한계가 있다. 반면 RGB 방식은 다양한 정보를 각 채널에 분산시켜 표현력이 뛰어나고 시각적 해석에 유리하나, 전처리 복잡도와 연산량이 증가한다. 이러한 차이는 모델의 크기, 학습 시간, 하드웨어 요구사항에도 영향을 미친다. 자원이 제한된 IoT·모바일 환경에서는 Grayscale 기반의 경량 모델이 적합할 수 있고, 반대로 복잡한 행위 기반 탐지에는 RGB 방식이 더 나은 성능을 낼 수 있다.

Ⅳ. 악성 트래픽 탐지 기법

이미지로 변환된 네트워크 트래픽은 딥러닝 기반 모델을 통해 정상 트래픽과 악성 트래픽으로 분류된다. 이러한 탐지 모델은 구조에 따라 단일 모델과 하이브리드 모델로 구분된다. 단일 모델은 하나의 딥러닝 아키텍처를 사용하는 방식으로, 구현이 간단하고 학습 속도가 빠르다는 장점이 있다. 반면, 하이브리드 모델은 두 개 이상의 모델 또는 분석 기법을 결합하여 탐지 성능을 향상시키고, 다양한 유형의 변형 공격에 보다 정교하게 대응할 수 있다는 강점을 지닌다. 또한, 제로데이 공격 대응 가능 여부에 따라 분류 방식은 두 가지로 나뉜다. Closed-set Recognition (CSR)은 학습된 클래스에 대해서만 분류를 수행하는 반면, Open-set Recognition (OSR)은 학습되지 않은 데이터를 'unknown' 클래스로 분류할 수 있어 제로데이 탐지에 활용될 수 있다. 본 장에서는 기존 연구들을 단일 모델, 하이브리드 모델, 제로데이 공격 대응 모델로 분류하고, 각 탐지 기법의 구조, 이미지화 방식, 입력 데이터, 모델 구성, 성능 특징 등을 체계적으로 분석한다.

4.1 단일 모델 기반 탐지 기법

단일 모델 기반 탐지 기법은 하나의 딥러닝 모델을 사용해 악성 트래픽을 분류하는 방식으로, 주로 CNN 기반 구조가 활용된다. 이 방식은 크게 (1) 기존 이미지 분류 모델을 그대로 적용하는 접근과, (2) 트래픽 이미지의 특성에 맞춰 맞춤형 CNN 구조를 설계하는 방식으로 나눌 수 있다. 예를 들어, [20]은 트래픽을 MNIST와 동일한 28×28 Grayscale 이미지로 변환하여 LeNet-5 모델을 적용하였으며, [23] 역시 23×23 Grayscale 이미지에 LeNet-5 유사 구조를 사용하였다. [24]는 모바일 트래픽을 227×227×3 Grayscale 이미지로 변환하여 AlexNet 구조에 적용함으로써 높은 분류 정확도를 달성하였다. [25]는 트래픽을 50×50 RGB 이미지로 변환한 뒤, VGG 구조를 간소화한 CNN 모델을 사용하여 분류하였고, [26]은 Grayscale 이미지로 변환

표 2. 방법론 비교
Table 2. Methodology Comparison

Method	Dataset	Target	Image Type	Classification Type*			Accuracy	F1-score
				Binary	Multi	OSR†		
[20]	USTC-TFC'16	Malware	Grayscale (28*28)		✓(20§)		99.17	99.97
[21]	Own Dataset	Malware	RGB (-)	✓			91.32	91.35
[22]	UNSW-NB15	Malicious Traffic	Grayscale (28*28)	✓			96.82	98.52
[23]	CICIoT'23	Malicious Traffic	Grayscale (23*23)	✓			99.71	98.47
[24]	CICAndMal'17	Malware	Grayscale (227*227)	✓			-	99.97
[25]	MCFP, USTC-TFC'16, MedBIoT, IEEE-Mirai	Malware	RGB (50*50)		✓(142)		-	97.00
[26]	CTU-13, ISOT HTTP Botnet Dataset, Own Dataset	Malicious Traffic	Grayscale (-)	✓			99.98	99.98
[27]	Own Dataset	Malware	Grayscale (64*64)	✓			99.48	85.86
[28]	USTC-TFC'16, ISCX-Tor'16	Malicious Traffic	Grayscale (28*28)		✓(10§)		95.11	94.98
[29]	CICAndMal'17	Malware	Grayscale (28*28)	✓			99.19	98.70
[30]	CICIDS'17	Malicious Traffic	Grayscale (40*40)		✓(11)		99.81	99.91
	CTU-13			✓			99.82	99.87
[31]	CICAndMal'17, CICInvesAndAml'19	Malware	Grayscale (28*28)	✓	✓(4)		99.98(bin) 99.99(multi)	99.98(bin) 99.99(multi)
[32]	CICIDS'17	Malicious Traffic	Grayscale (16*16)		✓(12)		99.87	99.79
[33]	MCFP	Malware	RGB (16*16)		✓(12)		99.07	96.25
[34]	IoT-23	Malware	RGB (224*224)		✓(4§)		93.00	91.00
[35]	CTU-13, MCFP	Malware	Grayscale (28*28)	✓			99.90	99.90
[36]	CTU-13, CICIDS'17	Malicious Traffic	Grayscale (40*40)		✓(9)	✓(1)	85.83	82.90
[37]	CTU-13, CICIDS'17	Malicious Traffic	Grayscale (40*40)		✓(9)	✓(1)	85.83	82.90
[38]	CICIDS'17	Malicious Traffic	Grayscale (-)	✓		✓(1)	83.93	90.69
	Own Dataset						96.62	83.66
	HMCT-2020						-	98.66
[39]	ISCX-VPN-Tor	Malicious Traffic	Grayscale (32*32)		✓(13§)	✓(12§)	88.60	81.38
	NSL-KDD				✓(10)	✓(7)	91.88	93.66
	Own Dataset				✓(5§)	✓(3)	96.25	97.43

* (n) indicates the number of classes used in each classification type (e.g., multi-class or OSR)
† denotes that the method incorporates open-set recognition
§ represents that the classification includes the normal class among the multiple classes

한 봇넷 트래픽에 DenseNet을 적용하여 사전학습 없이도 우수한 성능을 보였다. 한편, [22], [27], [28]은 기존 CNN 구조를 그대로 사용하는 대신, 입력 이미지 크기와 트래픽 특성에 최적화된 맞춤형 아키텍처를 설계하였다. 특히 [28]은 암호화된 트래픽 분석을 위해 28×28 Grayscale 이미지를 입력으로 사용하고, 일반적인 Convolution 대신 PConv 및 Do-Conv 연산을 적용한 경량 모델인 FasterTrafficNet을 제안하였다.

4.2 하이브리드 모델 기반 탐지 기법

하이브리드 모델 기반 탐지 기법은 서로 다른 딥러닝 구조나 분석 기법을 결합함으로써 탐지 성능을 향상시키고, 시공간적 특징을 동시에 반영하는 접근이다. [29], [31]은 악성 안드로이드 앱 탐지를 위해 정적 분석과 동적 분석을 결합한 2단계 탐지 모델을 제안하였다. 먼저 앱 구성 정보를 기반으로 정적 분석을 수행한 후, 네트워크 트래픽을 이미지로 변환하여 동적 분석을 수행하는 구조다. [29]은 정적 분석에 FCNN을 적용해 앱의 정상 여부를 판별하고, 정상으로 분류된 앱의 트래픽만 28×28 Grayscale 이미지로 변환해 CAE 및 CNN 기반 동적 분석을 수행하였다. [31]은 정적 분석에 Random Forest를 활용하고, 신뢰도가 낮은 트래픽에 대해 28×28 Grayscale 이미지로 변환한 후, CNN과 LSTM을 병렬 구조로 구성하여 공간적·시간적 특징을 동시에 학습하였다. 반면 [30]은 LeNet-5 기반 CNN과 LSTM을 직렬로 연결한 구조를 사용하였다. 이 모델은 트래픽을 40×40 Grayscale 이미지로 변환한 후, CNN의 출력 벡터를 시퀀스로 간주해 LSTM에 전달함으로써 시공간 정보를 순차적으로 학습한다. 두 연구는 CNN과 LSTM 간 결합 방식에 따라 시공간 정보 활용 방식에서 차이를 보인다.

데이터셋의 클래스 불균형 문제를 해결하기 위한 연구도 활발히 이루어지고 있다. [32]는 다수 클래스를 언더샘플링하고, 16×16 Grayscale 이미지로 변환한 후, Inception과 ResNet을 결합한 RICNN 모델을 제안하였다. Inception 유닛은 다양한 커널 크기의 병렬 구조로 소수 클래스의 공간적 특징을 효과적으로 추출하고, ResNet의 잔차 연결은 기울기 소실을 완화하여 학습 후반에도 소수 클래스 정보가 유지되도록 한다. 반면, [33]은 GAN을 활용해 소수 클래스의 트래픽 이미지를 보완하는 방식을 제안하였다. 트래픽을 16×16 RGB 이미지로 변환한 뒤, GAN으로 새로운 이미지를 생성하고 유의미한 샘플만 선별하여 ResNet 기반 CNN으로 분류하였다.

Few-shot 학습 기반 접근으로는 [34]가 Prototypical GNN(PGNN)을 활용한 모델을 제안하였다. 트래픽을 256×256 RGB 이미지로 변환하고, 사전 학습된 ResNet-18을 통해 고차원 특징 벡터를 추출한 뒤, 이를 GCN의 노드 임베딩으로 사용하였다. 이미지 간 유사도를 기반으로 그래프를 구성하고, GCN을 통해 노드 간 관계를 학습함으로써 제한된 학습 데이터 환경에서도 효과적인 탐지를 수행할 수 있다. 마지막으로, [35]는 암호화된 트래픽의 구조적 특징과 세션 정보를 통합하는 멀티뷰 GNN 모델을 제안하였다. 세션 단위 트래픽을 28×28 Grayscale 이미지로 변환하고 이를 784차원 벡터로 펼쳐, 유클리디안 거리 기반 KNN 그래프를 구성하였다. 세션 메타데이터는 별도로 벡터화되어 그래프의 속성으로 활용되며, 최종적으로 GraphSAGE를 통해 학습이 이루어진다. 이를 통해 암호화된 트래픽에서도 높은 탐지 성능을 달성하였다.

4.3 제로데이 공격 탐지 모델

기존의 대부분 연구는 학습된 클래스 내에서만 분류가 가능한 CSR(Closed-Set Recognition) 방식에 기반하고 있어, 제로데이 공격과 같은 미지 클래스에 효과적으로 대응하기 어렵다. [21], [24]는 일부 알려지지 않은 공격을 정상 또는 악성으로 분류하는 실험을 수행하였으나, 해당 트래픽을 별도의 unknown 클래스로 식별하지는 못했다. 이에 따라 최근에는 OSR(Open-Set Recognition)을 적용한 연구들이 등장하고 있다. [36]은 CNN의 출력층에 Softmax 대신 OpenMax를 적용한 Open-CNN 모델을 제안하였다. 이 모델은 40×40 Grayscale 이미지로 변환된 트래픽을 입력으로 사용하고, 입력 데이터가 각 클래스 중심으로부터 벗어난 정도를 기반으로 unknown 여부를 판단하여 분류한다. [37]은 이 모델에 Active Learning을 결합하여, Open-CNN이 unknown으로 분류한 샘플 중 불확실성이 높은 데이터만 선별해 최소한의 라벨링으로 확장 학습을 수행함으로써 학습 효율을 높였다.

하이브리드 구조 기반의 OSR 모델도 제안되고 있다. [38]은 기존 연구[33]처럼 소수 클래스의 트래픽 이미지를 증강하는 방식 대신, GAN을 이용해 HTTP 기반의 악성 플로우를 직접 생성하였다. 제안된 HMCD-Model은 CNN을 통해 이미지의 공간적 특징을 추출하고, LSTM을 통해 시계열 관계를 학습하여 트래픽의 시공간적 특성을 반영한다. 이를 통해 unknown 트래픽에 대해서도 정상 또는 악성 여부를 효과적으로 판별하였다. [39]는 트래픽을 30×30 Grayscale 이미지로 변환하고, CNN(AlexNet, VGG16)으로 공간적 특징을, LSTM으로 시간적 특성을 각각 추출하였다. 이후

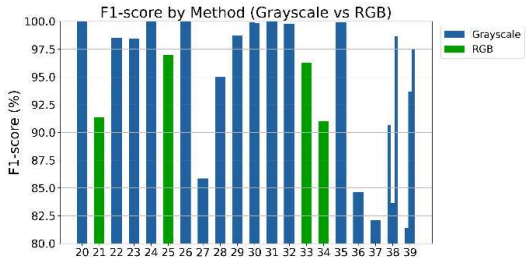

그림 2. 이미지 변환 방식별 성능 비교
Fig. 2. Performance by Image Conversion

mRMR 알고리즘을 사용해 상위 100개의 주요 특징만을 선별하여 SVM 분류기에 입력하고, 밀도 기반 클러스터링(DBSCAN)을 결합함으로써 unknown 트래픽의 이상 여부를 탐지하였다. 특히 이 모델은 점진적 학습 구조를 도입하여 새로운 유형의 공격이 등장하더라도 지속적으로 탐지 성능을 유지할 수 있음을 실험적으로 입증하였다.

V. 기존 연구 분석 및 한계점

기존 연구들은 네트워크 트래픽을 Grayscale 또는 RGB 이미지로 변환한 후, 딥러닝 모델을 활용해 악성 행위를 탐지하는 다양한 기법을 제안해왔다. 본 장에서는 이러한 연구들을 성능과 실용성 관점에서 분석하고, 한계점을 바탕으로 향후 연구 방향을 제시한다.

먼저, IoT 환경을 대상으로 한 연구들[21,23,26,34,36,37]은 CPU 및 메모리 자원이 제한된 상황에서 경량성과 실시간성 확보가 핵심 과제로 제시된다. 이를 위해 많은 연구에서 모델 구조 단순화나 사전 학습 모델 활용을 통해 경량화를 시도하였으나, 실시간성에 대한 정량적 평가는 일부 연구[23,26,37]에만 제한적으로 보고되었고, 대부분은 개별 샘플 처리 시간조차 명시하지 않았다. 또한 IoT 기기 간 통신 프로토콜의 다양성을 고려할 때, 탐지 기법의 프로토콜 독립성 확보도 중요한 과제다. [21]은 이미지 기반 분석이 프로토콜 필드에 의존하지 않기 때문에 프로토콜에 독립적이라고 주장했고, [23]은 트래픽만으로 기기 종류를 구별할 수 있음을 실험으로 입증했다. 그럼에도 불구하고 실시간성, 경량성, 프로토콜 다양성에 대한 종합적이고 체계적인 성능 검증은 여전히 부족하다.

OSR을 적용한 연구는 CSR 기반 연구에 비해 상대적으로 적으며, 적용된 경우에도 대부분 미지 데이터를 단일 unknown 클래스로 분류하는 수준에 머무르고 있다[36-39]. 이로 인해 세부 공격 유형 식별이 어려워 여전히 수동적인 사후 분석이 요구된다. [39]는 unknown 트래픽을 세부 유형으로 분류하고, 새로운 클래스를 지속적으로 학습할 수 있는 구조를 제안했지만, 전통적인 CSR 대비 탐지 정확도가 낮다는 한계를 보였다. 그림 2에서도 OSR 기반 탐지 기법 전반의 성능이 낮게 나타난다. F1-score가 99% 이상인 연구들[20,24,26,30,32,35]은 공통적으로 Grayscale 기반 방식을 사용하였으며, 성능이 낮은 연구들[21,25,33,34]은 데이터셋 구성 또는 모델 구조의 한계를 원인으로 지적하였다. [26]은 이미지 생성 과정에서 시간 정보가 누락되어 트래픽의 타이밍 특성을 반영하지 못한다는 점에서 이미지화 기법의 구조적 제약을 지적하였다. 그러나 다수의 연구는 전처리 방식이나 이미지화 전략이 탐지 성능에 미치는 영향을 체계적으로 분석하지 않았다.

이러한 분석을 바탕으로 향후 연구는 다음과 같은 방향으로 확장될 필요가 있다. 첫째, IoT 환경에 적합한 경량 모델 구조 개발과 함께 실시간성에 대한 정량적이고 일관된 평가가 요구된다. 둘째, 다양한 통신 프로토콜 환경에서도 안정적으로 동작할 수 있는 탐지 기법의 프로토콜 독립성을 실험적으로 입증해야 한다. 셋째, OSR 기반 기법의 일반화 성능을 향상시키고 실제 환경에서의 성능을 평가할 수 있는 정교한 OSR 모델 개발이 필요하다. 마지막으로, Grayscale과 RGB 방식 간 성능 비교, 전처리 및 이미지화 전략이 탐지 성능에 미치는 영향을 실험적으로 분석하고, 트래픽 유형별로 최적화된 이미지 변환 방식을 제안하는 연구로의 확장이 요구된다.

VI. 결 론

본 논문은 네트워크 트래픽을 이미지로 변환한 뒤 딥러닝 모델을 활용해 악성 행위를 탐지하는 연구 동향을 체계적으로 분석하였다. 기존 연구들을 모델 구조, 이미지 처리 방식, 탐지 대상, 실험 환경 등의 기준에 따라 분류·비교하고, 각 접근 방식의 특징과 차이점을 정리하였다. 이미지화 기법은 Grayscale과 RGB 방식으로 나뉘며, 전처리된 트래픽 데이터를 시각적 형태로 변환해 딥러닝 모델의 입력으로 활용한다. 탐지 모델은 CNN 기반 구조에서 시작해 시공간 정보를 반영한 CNN-LSTM, 이미지 생성을 활용한 GAN, 그래프 기반의 GNN 등으로 확장되고 있다. 또한, 알려지지 않은 공격을 탐지하기 위한 Open-set Recognition 기반 제로데이 탐지 기법에 대한 연구도 점차 활발히 진행되고 있다. 본 논문은 이러한 연구 흐름을 종합적으로 정리함으로써, 이미지 기반 악성 트래픽 탐지 분야의 이해를

돕고 향후 연구 설계에 유용한 기초 자료로 활용될 수 있을 것이다.

References

[1] T. Bouraffa and K. L. Hui, “Regulating information and network security: Review and challenges,” *ACM Comput. Surv.*, vol. 57, no. 5, pp. 1-38, 2025. (https://doi.org/10.1145/3711124)

[2] H. Tahaei, F. Afifi, A. Asemi, F. Zaki, and N. B. Anuar, “The rise of traffic classification in IoT networks: A survey,” *J. Netw. Comput. Appl.*, vol. 154, Article 102538, 2020. (https://doi.org/10.1016/j.jnca.2020.102538)

[3] Y. Bengio, A. Courville, and P. Vincent, “Representation learning: A review and new perspectives,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798-1828, 2013. (https://doi.org/10.1109/TPAMI.2013.50)

[4] A. Alenazi, I. Traore, K. Ganame, and I. Woungang, “Holistic model for HTTP botnet detection based on DNS traffic analysis,” in *Proc. Int. Conf. Intell., Secure, and Dependable Syst. in Distributed and Cloud Environments (ISDDC 2017)*, pp. 1-18, Vancouver, Canada, Oct. 2017. (https://doi.org/10.1007/978-3-319-69155-8_1)

[5] García, S., & Uhlir, V. *Malware capture facility project*(2013), Apr. 14, 2025, http://mcfp.felk.cvt.cz.

[6] A. H. Lashkari, A. F. A. Kadir, L. Taheri, and A. A. Ghorbani, “Toward developing a systematic approach to generate benchmark Android malware datasets and classification,” in *Proc. 52nd IEEE Int. Carnahan Conf. Security Technol. (ICCST)*, Montreal, QC, Canada, 2018. (https://doi.org/10.1109/CCST.2018.8585560)

[7] L. Taheri, A. F. A. Kadir, and A. H. Lashkari, “Extensible android malware detection and family classification using network-flows and API-calls,” in *Proc. 2019 Int. Carnahan Conf. Security Technol. (ICCST)*, pp. 1-8, Oct. 2019. (https://doi.org/10.1109/CCST.2019.8888430)

[8] S. Mahdavifar, A. F. A. Kadir, R. Fatemi, D. Alhadidi, and A. A. Ghorbani, “Dynamic android malware category classification using semi-supervised deep learning,” in *Proc. 2020 IEEE Int. Conf. Dependable, Autonomic and Secure Comput. (DASC)*, pp. 515-522, Aug. 2020. (https://doi.org/10.1109/DASC-PICom-BDCom-CyberSciTech49142.2020.00094)

[9] M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, “A detailed analysis of the KDD CUP 99 data set,” in *Proc. 2nd IEEE Symp. Comput. Intell. Security Defense Appl. (CISDA)*, pp. 1-6, 2009. (https://doi.org/10.1109/CISDA.2009.5356528)

[10] A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, “Toward developing a systematic approach to generate benchmark datasets for intrusion detection,” *Comput. Secur.*, vol. 31, no. 3, pp. 357-374, May 2012. (https://doi.org/10.1016/j.cose.2011.12.012)

[11] S. Garcia, M. Grill, J. Stiborek, and A. Zunino, “An empirical comparison of botnet detection methods,” *Comput. Secur.*, vol. 45, pp. 100-123, 2014. (https://doi.org/10.1016/j.cose.2014.05.011)

[12] N. Moustafa and J. Slay, “UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set),” in *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, IEEE, 2015. (https://doi.org/10.1109/MilCIS.2015.7348942)

[13] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, “Toward generating a new intrusion detection dataset and intrusion traffic characterization,” in *Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy (ICISSP)*, Portugal, Jan. 2018. (https://doi.org/10.5220/0006639801080116)

[14] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, “Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy,” in *Proc. 53rd IEEE Int. Carnahan Conf. Secur. Technol.*, Chennai,

India, 2019.
(https://doi.org/10.1109/CCST.2019.8888419)

[15] S. Dadkhah, H. Mahdikhani, P. K. Danso, A. Zohourian, K. A. Truong, and A. A. Ghorbani, "Towards the development of a realistic multidimensional IoT profiling dataset," in *Proc. 19th Annu. Int. Conf. Privacy, Secur. Trust (PST)*, Fredericton, Canada, Aug. 2022.
(https://doi.org/ 10.1109/PST55820.2022.9851966)

[16] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," *Sensors*, vol.23, no. 13, Article 5941, 2023.
(https://doi.org/10.3390/s23135941)

[17] G. D. Gil, A. H. Lashkari, M. Mamun, and A. A. Ghorbani, "Characterization of encrypted and VPN traffic using time-related features," in *Proc. 2nd Int. Conf. Inf. Syst. Secur. Privacy (ICISSP)*, pp. 407-414, Rome, Italy, 2016.
(https://doi.org/ 10.5220/0005740704070414)

[18] G. Conti, S. Bratus, S. Shubina, A. Lichtenberg, A. Ragsdale, R. Perez-Alemany, R. Sangster, and M. Supan, "A visual study of binary fragment types," in Proc. *Black Hat USA*, 2010.

[19] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware images: Visualization and automatic classification," in *Proc. 8th Int. Symp. Vis. Cyber Secur.*, pp. 1-7, Jul. 2011.
(https://doi.org/10.1145/2016904.201690)

[20] W. Wang, M. Zhu, X. Zeng, X. Ye, and Y. Sheng, "Malware traffic classification using convolutional neural network for representation learning," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, pp. 712-717, Jan. 2017.
(https://doi.org/ 10.1109/ICOIN.2017.7899588)

[21] R. Shire, S. Shiaeles, K. Bendiab, B. Ghita, and N. Kolokotronis, "Malware squid: A novel IoT malware traffic analysis framework using convolutional neural network and binary visualisation," in *Proc. 19th Int. Conf. Internet Things, Smart Spaces, Next Gener. Netw. Syst.*, pp. 65-76, St. Petersburg, Russia, Aug. 2019.
(https://doi.org/10.1007/978-3-030-30859-9_6)

[22] Y. Wang, J. An, and W. Huang, "Using CNN-based representation learning method for malicious traffic identification," in *Proc. IEEE/ACIS 17th Int. Conf. Comput. Inf. Sci. (ICIS)*, pp. 400-404, Jun. 2018.
(https://doi.org/ 10.1109/ICIS.2018.8466404)

[23] M. Hamidouche, E. Popko, and B. Ouni, "Enhancing IoT security via automatic network traffic analysis: The transition from machine learning to deep learning," in *Proc. 13th Int. Conf. Internet Things*, pp. 105-112, Nov. 2023.
(https://doi.org/10.1145/3627050.3627053)

[24] J. Kang and S. Lee, "Android malware detection through the conversion of network traffic to image," *J. KIISE*, vol. 47, no. 8, pp. 761-768, 2020.
(https://doi.org/10.5626/JOK.2020.47.8.761)

[25] R. E. Davis, J. Xu, and K. Roy, "Classifying malware traffic using images and deep convolutional neural network," *IEEE Access*, 2024.
(https://doi.org/10.1109/ACCESS.2024.3391022)

[26] S. Taheri, M. Salem, and J. S. Yuan, "Leveraging image representation of network traffic data and transfer learning in botnet detection," *Big Data Cogn. Comput.*, vol. 2, no. 4, Article 37, 2018.
(https://doi.org/10.3390/bdcc2040037)

[27] F. A. Demmese, A. Neupane, S. Khorsandroo, M. Wang, K. Roy, and Y. Fu, "Machine learning based fileless malware traffic classification using image visualization," *Cybersecurity*, vol. 6, no. 1, Article 32, 2023.
(https://doi.org/10.1186/s42400-023-00170-z)

[28] L. Yu, J. Yuan, J. Zheng, and N. Yang, "A model of encrypted network traffic classification that trades off accuracy and efficiency," *J. Netw. Syst. Manage.*, vol. 33, no. 1, pp. 1-32, 2025.
(https://doi.org/10.1007/s10922-024-09892-y)

[29] J. Feng, L. Shen, Z. Chen, Y. Wang, and H. Li, “A two-layer deep learning method for android malware detection using network traffic,” *IEEE Access*, vol. 8, pp. 125786-125796, 2020. (https://doi.org/10.1109/ACCESS.2020.3008081)

[30] Y. Zhang, X. Chen, L. Jin, X. Wang, and D. Guo, “Network intrusion detection: Based on deep hierarchical network and original flow data,” *IEEE Access*, vol. 7, pp. 37004-37016, 2019. (https://doi.org/10.1109/ACCESS.2019.2905041)

[31] C. Ding, N. Luktarhan, B. Lu, and W. Zhang, “A hybrid analysis-based approach to android malware family classification,” *Entropy*, vol. 23, no. 8, Article 1009, 2021. (https://doi.org/10.3390/e23081009)

[32] B. Xia, D. Han, X. Yin, and G. Na, “RICNN: A ResNet&Inception convolutional neural network wjfor intrusion detection of abnormal traffic,” *Comput. Sci. Inf. Syst.*, vol. 19, no. 1, pp. 309-326, 2022. (https://doi.org/10.2298/CSIS210617055X)

[33] X. Cao, Q. Luo, and P. Wu, “Filter-GAN: Imbalanced malicious traffic classification based on generative adversarial networks with filter,” *Mathematics*, vol. 10, no. 19, Article 3482, 2022. (https://doi.org/10.3390/math10193482)

[34] T. T. Thein, Y. Shiraishi, and M. Morii, “Few-shot learning-based malicious IoT traffic detection with prototypical graph neural networks,” *IEICE Trans. Inf. Syst.*, vol. 106, no. 9, pp. 1480-1489, 2023. (https://doi.org/10.1587/transinf.2022OFP0004)

[35] Y. Hong, Q. Li, Y. Yang, and M. Shen, “Graph based encrypted malicious traffic detection with hybrid analysis of multi-view features,” *Inf. Sci.*, vol. 644, Article 119229, 2023. (https://doi.org/10.1016/j.ins.2023.119229)

[36] Y. Zhang, J. Niu, D. Guo, Y. Teng, and X. Bao, “Unknown network attack detection based on open set recognition,” *Procedia Comput. Sci.*, vol. 174, pp. 387-392, 2020. (https://doi.org/10.1016/j.procs.2020.06.104)

[37] Z. Zhang, Y. Zhang, J. Niu, and D. Guo, “Unknown network attack detection based on open-set recognition and active learning in drone network,” *Trans. Emerg. Telecommun. Technol.*, vol. 33, no. 10, Article e4212, 2022. (https://doi.org/10.1002/ett.4212)

[38] X. Yun, J. Xie, S. Li, Y. Zhang, and P. Sun, “Detecting unknown HTTP-based malicious communication behavior via generated adversarial flows and hierarchical traffic features,” *Comput. Secur.*, vol. 121, Article 102834, 2022. (https://doi.org/10.1016/j.cose.2022.102834)

[39] J. Liu, J. Wang, T. Yan, F. Qi, and G. Chen, “Unknown traffic recognition based on multi-feature fusion and incremental learning,” *Appl. Sci.*, vol. 13, no. 13, Article 7649, 2023. (https://doi.org/10.3390/app13137649)

박 영 주 (Young-joo Park)

2022년 3월~현재 : 서울여자대학교 정보보학과 재학

<관심분야> AI 보안, 네트워크 보안, 개인정보 보호

[ORCID:0009-0000-6008-1840]

이 선 우 (Sun Woo Lee)

2015년 2월 : 서강대학교 수학 학사

2022년 2월 : 고려대학교 정보보호대학원 박사

2022년 3월~9월 : 고려대학교 정보보호대학원 연구교수

2022년 10월~2024년 2월 : Samsung Research Staff Engineer

2024년 3월~현재 : 서울여자대학교 지능정보보호학부 조교수

<관심분야> Usable security, IoT security, Side-channel attack, Privacy leakage attack

[ORCID:0000-0001-5216-0266]