



NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

**OVERCOMING TOR: AN ANALYSIS
OF VIDEO FINGERPRINTING ATTACKS
WITH MACHINE LEARNING**

by

Trevor J. Thomas

September 2023

Thesis Advisor:
Second Reader:

Armon C. Barton
Geoffrey G. Xie

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE September 2023	3. REPORT TYPE AND DATES COVERED Master's thesis		
4. TITLE AND SUBTITLE OVERCOMING TOR: AN ANALYSIS OF VIDEO FINGERPRINTING ATTACKS WITH MACHINE LEARNING			5. FUNDING NUMBERS	
6. AUTHOR(S) Trevor J. Thomas				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.			12b. DISTRIBUTION CODE A	
13. ABSTRACT (maximum 200 words) Terrorist groups and violent extremist organizations have taken advantage of the anonymity provided by The Onion Router (Tor). They use it to clandestinely distribute video media to recruit and spread propaganda as well as incite others to commit acts of terrorism. This thesis seeks to continue a line of work in developing machine learning algorithms to conduct video fingerprinting attacks on network traffic transmitted over Tor in a closed world environment. It will expand upon previous research by using multiple media platforms for videos being streamed. The end goal from this line of research is a future product that the Department of Defense can apply against real world network traffic to identify individuals using Tor and the dark web for recruiting, training, and incitement of terror as well as identify individuals that are actively being recruited.				
14. SUBJECT TERMS video fingerprinting, AI, TOR, artificial intelligence, The Onion Router			15. NUMBER OF PAGES 49	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

**OVERCOMING TOR:
AN ANALYSIS OF VIDEO FINGERPRINTING
ATTACKS WITH MACHINE LEARNING**

Trevor J. Thomas
Lieutenant, United States Navy
BSAS, University of South Florida, 2016

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

**NAVAL POSTGRADUATE SCHOOL
September 2023**

Approved by: Armon C. Barton
Advisor

Geoffrey G. Xie
Second Reader

Gurminder Singh
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Terrorist groups and violent extremist organizations have taken advantage of the anonymity provided by The Onion Router (Tor). They use it to clandestinely distribute video media to recruit and spread propaganda as well as incite others to commit acts of terrorism. This thesis seeks to continue a line of work in developing machine learning algorithms to conduct video fingerprinting attacks on network traffic transmitted over Tor in a closed world environment. It will expand upon previous research by using multiple media platforms for videos being streamed. The end goal from this line of research is a future product that the Department of Defense can apply against real world network traffic to identify individuals using Tor and the dark web for recruiting, training, and incitement of terror as well as identify individuals that are actively being recruited.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

1	Introduction	1
1.1	Purpose and Scope	2
1.2	Thesis Organization	2
2	Background and Related Work	3
2.1	Dark Web	3
2.2	Tor	4
2.3	How Tor works	5
2.4	Website Fingerprinting	7
2.5	Video Fingerprinting	8
2.6	Machine Learning Methods	10
2.7	Related Work	10
3	Methodology	11
3.1	Data Collection	11
3.2	The Crawler	14
3.3	The PCAP Parser	14
3.4	The Threat Model	15
3.5	Lab Environment	16
4	Implementation and Design	17
4.1	Data Collection	17
4.2	Feature Selection	17
4.3	Train-Test Split	18
4.4	Comparison to Previous Work and Results	19
5	Future Work and Conclusion	31
5.1	Future Work	31

5.2 Conclusion.	32
List of References	33
Initial Distribution List	35

List of Figures

Figure 2.1	Key exchange in Tor	6
Figure 3.1	Visualization of network trace.	16
Figure 4.1	Video ID prediction accuracy comparison	20
Figure 4.2	Video collection prediction accuracy	21
Figure 4.3	Genre prediction using last three videos as test set.	22
Figure 4.4	Confusion matrix for YouTube Genre prediction.	23
Figure 4.5	Confusion matrix for YouTube Genre prediction with color density indicating correct labels.	24
Figure 4.6	Confusion matrix for Vimeo Genre prediction.	25
Figure 4.7	Confusion matrix for Vimeo Genre prediction with color density indicating correct labels.	25
Figure 4.8	Confusion matrix for Rumble Genre prediction.	26
Figure 4.9	Confusion matrix for Rumble Genre prediction with color density indicating correct labels.	26
Figure 4.10	Confusion matrix for Facebook Genre prediction.	27
Figure 4.11	Confusion matrix for Facebook Genre prediction with color density indicating correct labels.	27
Figure 4.12	Genre prediction using first three videos as test set	28
Figure 4.13	Confusion matrix for YouTube Genre prediction using first three videos as test set.	29
Figure 4.14	Confusion matrix for YouTube Genre prediction using first three videos with color density indicating correct labels.	29

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

API	Application Program Interface
CSS	Cascading Style Sheets
CSV	Comma Separated Values
GPU	Graphics Processing Unit
HTML	HyperText Markup Language
ID	Identifier
ISP	Internet Service Provider
LTE	Long Term Evolution
NPS	Naval Postgraduate School
NUC	Next Unit of Computing
PCAP	Packet Capture
SSH	Secure Shell Protocol
Tor	The Onion Router
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
URL	Uniform Resource Locator
USN	U.S. Navy
VOIP	Voice Over Internet Protocol

VPN Virtual Private Network

WF Website Fingerprinting

CHAPTER 1:

Introduction

Cybersecurity remains a focal point for national security as it can be leveraged by adversaries to either strengthen their attacks or circumvent our defenses. A key attribute in cyber warfare is the ability to remain undetected, this is important whether you are an attacker taking down a website, a terrorist posting recruitment propaganda or a defender using a honeypot to learn an adversary's tactics.

There are many methods and tools to make it harder to discern the true origin of a cyber actor. One such method is the use of encrypted networks like The Onion Router (Tor) network. Encrypted networks provide security by taking readable text and converting it into an unreadable format that requires a key to decode it. The Tor Network has seen steady growth since its creation and remains a strong option for anyone wanting to help achieve anonymity due to its effectiveness and the fact that it is open source and free. Tor has been used by journalists, populations facing persecution, whistle blowers, terrorists, and for-profit cyber criminals. The "Silk Road," an online marketplace for drugs and other illicit activity, was accessible only by Tor.

Tor is not a one-stop shop for complete secrecy as it is vulnerable to traffic analysis. Emerging attacks utilizing machine learning, such as website fingerprinting (WF) attacks, have been shown to be effective. If an attacker can gain access to a user's web activity while they are using low latency networks, such as a virtual private network (VPN), they can leverage features within the network traffic such as time traffic was sent, size, and others in machine learning which can then be used to determine if a certain website was visited. Sirinam et al. [1] states that "state of the art website fingerprinting attacks have been shown to be effective against Tor." Streaming protocols have been known to leak information about the content that is streamed due to the network traffic patterns used by streaming services. Schuster et al. [2] developed a method using deep learning to exploit such leaks to be able to identify the video that is being streamed.

1.1 Purpose and Scope

The dataset that is being used in this research contains 10 videos each from six different categories for four video streaming services: YouTube, Vimeo, Rumble, and Facebook video. Each video is between three and six minutes in length and will have been streamed about 900 times each, a notable increase from previous studies. This research is using a dataset of encrypted network traffic collected from a simulated user who is using the Tor network and streaming videos. We use the term undefended to define network traffic that has not had any other proposed methods of defense for Tor applied, and is merely network traffic captured from a user that is using Tor. This research seeks to compare accuracy results to those achieved by Rahman et al. [3], whose team showed that video traffic protected by Tor is fingerprintable. This research seeks to compare accuracy results on undefended network traffic that has not been modified from the four different platforms in a closed-world scenario. It also seeks to compare accuracy on our YouTube dataset to results achieved by Kim [4], Campuzano [5], and Duhe [6], against undefended datasets of YouTube videos. We believe our dataset will produce better results because it is larger, has more training data available and is a cleaner data set due to improvements made in the crawler. This research will answer the following two questions: will differing genres of videos impact a model's accuracy when classifying video category or a specific video? Is one platform's video streaming easier to fingerprint than others?

1.2 Thesis Organization

Chapter 2 provides background information on the Dark Web and a summary of the Tor network and how it works. Next, Website Fingerprinting is explained as a lead-in to Video Fingerprinting. Convolutional neural networks and how they are useful for video fingerprinting are then explained, as well as related previous work on the topic. Chapter 3 explains the methodology for obtaining the data used in this research as well as an analysis of the dataset. Chapter 4 describes the used deep learning models and hyperparameters in further detail and the effectiveness of each model in the experiments. Chapter 5 covers future research to be done in the field of video fingerprinting.

CHAPTER 2:

Background and Related Work

This chapter covers background literature related to the Tor network, the Dark Web, and machine learning models. It examines literature that aims to determine the website a user is visiting or the video content that is being displayed to the user.

2.1 Dark Web

Malicious actors use network encryption services such as TOR to help obfuscate their identities on the web. Technological advances have created what we call the Dark Web or Darknet. Biddle et al. [7] define a darknet as “a collection of networks and technologies that is not a separate physical network but an application and protocol layer riding on existing networks. An example is peer-to-peer file sharing.” Stopping the Dark Web wasn’t feasibly possible and would have been seen as an attack on technological improvements and infrastructure. Biddle et al. [7] states: “The dramatic rise in efficiency of the darknet can be traced back to general technological improvements in infrastructure areas, while attempts to fight it can be seen as efforts to deprive it of one or more of its infrastructure items.” As technology has improved over the years it has only gotten easier to create and maintain what can be viewed as the Dark Web and arguably it has gotten much harder to take down or even monitor. One explanation of the Dark Web by Gehl is “websites built with standard web technologies (HTML, CSS, server-side scripting languages, hosting software) that can be viewed with a standard web browser, such as Firefox or Chrome, which is routed through special routing software packages. I do not define these sites in terms of whether Google has crawled them (the ‘deep’ definition)” [8]. Other definitions play on the ethical connotations of “dark” and apply to a wide variety of things ethically and morally bad or otherwise illegal.

The Dark Web is most frequently known for the fact that it is used by malicious actors to sell hacks and otherwise illegal or unethical content, all while remaining anonymous. Samtani et al. state that “Cybersecurity experts have appraised the total annual cost of hacktivism, espionage, cyberwarfare and other cybercrime hacking activities against major

entities such as Equifax, Uber, and Yahoo! at 450 billion dollars” [9]. However, Samtani et al. go on to state that DarkNet marketplaces and forums tend to be more focused on selling weapons, drugs, or pornography than cybersecurity content [9]. Much of this content is not easily accessible on the web and a user will not find this content unless they are explicitly looking for it, have the direct I.P. address, and have been referred by word of mouth or via a forum where they are already known; what amounts to a buddy system to help hide the illicit content from authorities. Much of this content and many of the websites can only be accessed using a special web browser such as Tor.

2.2 Tor

Originally created in the mid-1990s by the United States Navy (USN) to protect communications, The Onion Router is now open source software that supports anonymous communications that is available to everyone. From April 2020 to April 2023 Tor went from an average of 2 million users to 3 million users in the span of three years and continues to steadily increase [10].

Dingledine et al. [11] explains some of the main advantages of Tor such as “Tor works on the real-world internet, requires no special privileges or kernel modifications, requires little synchronization or coordination between nodes, and provides a reasonable tradeoff between anonymity, usability, and efficiency.” Tor works by creating a distributed network that aims to anonymize traffic sent over Transmission Control Protocol (TCP). “Clients choose a path through the network and build a circuit, in which each node or ‘onion router’ in the path knows its predecessor and successor, but no other nodes in the circuit. Traffic flows down the circuit in fixed-size cells, which are unwrapped by a symmetric key at each node and relayed downstream” [11]. This is how Tor maintains secrecy: each hop new keys are created between the nodes the data is being transferred between and those are the only keys they have access to so no one node can intercept traffic from another node and be able to decrypt it.

This helped make Tor more robust as an initiating node would know when a circuit hop did not instantiate and could then try another new node. Tor also features what is known as leaky-pipe topology which can be used to avoid some attacks. Dingledine et al. explain leaky-pipe topology “through in-band signaling within the circuit, Tor initiators can direct

traffic to nodes partway down the circuit. This novel approach allows traffic to exit the circuit from the middle, possibly frustrating traffic shape and volume attacks based on observing the end of the circuit” [11]. Further enhancing security, Tor verifies the integrity of the data prior to it leaving the network to ensure no relay cell has altered the contents of the data being passed along. Tor has a simple goal in mind, anonymity that doesn’t heavily impact the user’s usability of the internet by maintaining low latency, high bandwidth, and ease of use. Tor does not take a peer-to-peer approach nor does it claim to protect against end-to-end attacks from an adversary that has the resources to look at all users connected to Tor [11]. Such an adversary can take advantage of timing and volume differences between users to infer which user is looking up what content.

2.3 How Tor works

Circuits built within the Tor network typically consist of a Guard node, Middle node and an Exit node, but this can be adjusted by the user. The Tor project maintains a list of all active relays that is kept up to date and pushed to all Tor clients every hour so that every client has the same information to build relays from.

Guard nodes are the first node in a circuit offering the guard the unique vantage point of being able to observe traffic coming from the user. A middle relay is simply the intermediate hop between the guard and exit relay. The selection process for a Guard node is more rigorous and has criteria such as bandwidth requirements. If a guard node is selected, a middle/exit node cannot be selected that is also owned by the owner of the guard node. While this does reduce the overall possible circuits, it helps prevent a malicious user, such as a nation state that can fund many nodes, from being able to potentially see all your traffic over Tor. Between the guard and exit relays is the middle relay which is used to provide better security than two relays can provide.

The exit relay is where traffic is decrypted and sent to its intended destination which will only see the exit relay IP address and not the IP address of the user that is using Tor. The owner of these relays will be served with the legal ramifications of the users’ actions whose traffic exited their exit node. Ideally these are handled by institutions and not out of homes for enhanced bandwidth capability and ability to respond to abuse complaints or law enforcement inquiries [12]. Another type of node is called a Bridge node. The Tor

networks list of IP addresses of relays is public information and as such it becomes easy for an authoritarian government to block all traffic to and from them. Bridge nodes are additional relays that are not listed publicly, making it harder for governments to block the use of Tor from their countries.

The Tor network is an overlay network. Figure 2.1 shows a basic example of how it works. Dingledine et al. explain the process below.

Each user runs local software called an onion proxy to fetch directories, establish circuits across the network, and handle connections from user applications. These onion proxies accept TCP streams and multiplex them across circuits and the onion router. The other side of the circuit connects to the requested destinations and relays the data. Each cell or onion router communicates with other cells and also onion proxies over Transport Layer Security (TLS), this secures the data with perfect forward secrecy and prevents an attacker from modifying data on the wire or impersonating an onion router. [11]

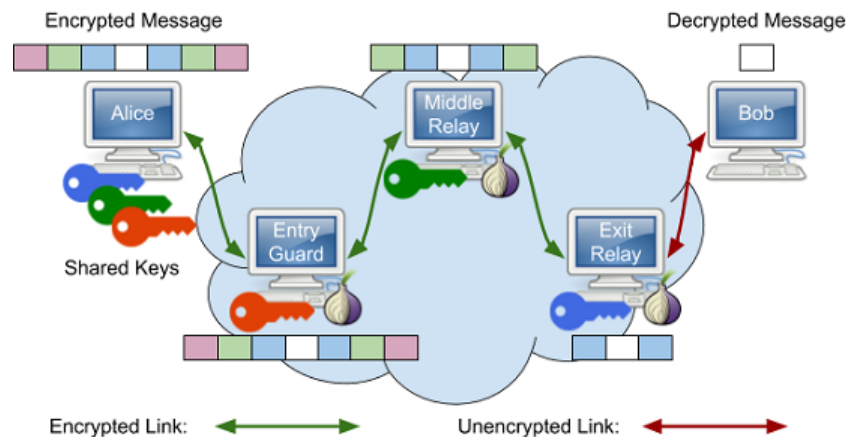


Figure 2.1. A visual depiction of encryption keys shared in the Tor network. Source: [13].

Tor uses fixed 512 byte cells comprised of a payload and a header which has the circuit

information and a description of what to do with the payload. They can be either control cells which the node that receives it will interpret or relay cells which carry data that will go to the exit node. Relay cells also have a header at the front to identify which datastream it is a part of (since one circuit can contain many datastreams multiplexed over it). A circuit is not limited to a single TCP stream and may contain multiple TCP streams. Circuits are expired and recreated periodically to help prevent linkability. This is done in advance to reduce time to create new circuits if circuit creation fails. Integrity of a stream is only performed at the beginning and end of a stream. The design is inherently vulnerable to an adversary observing on both ends, there is increased overhead at each relay, and onion routers cannot verify the data from intermediate hops, as they do not have the other onion router's session keys.

2.4 Website Fingerprinting

The term website fingerprinting was originally coined by Andrew Hintz [14] in 2002 when he showed that traffic analysis could be used against a web encryption proxy and allow someone monitoring that traffic to determine if a user was visiting certain websites. Separate TCP connections were created for each file that a web page loaded and due to the large number of them, most websites had a noticeable difference in connections created.

Encrypted connections do not provide full secrecy. Website fingerprinting can be used to determine the websites visited even over an encrypted connection. Sirinam et al. explain in their research that “attacks have been shown to be effective against Tor” [1]. Using the power of convolutional neural networks' deep learning ability, an attacker is able to get over 98 percent accuracy against Tor traffic that has no defenses [1]. Website fingerprinting allows an adversary to identify which web pages are being visited by detecting underlying patterns in network traffic and encrypted network traffic. Differences in content of websites such as images, ads, or scripts that are loaded can be deduced from encrypted traffic. This enables a passive adversary that can observe the network traffic, such as the Internet Service Provider or an individual that is sniffing a user's wireless connection, to be able to identify what websites a user has visited while using Tor to preserve his/her anonymity.

As website fingerprinting exploits the differences in content between websites it is a classification problem for machine learning. A classifier can be trained on a set of websites learning

the features that are unique to that network traffic and then an attack can be deployed that uses that classifier to match the user to one of the websites in the set. The classifier and set of features determines the effectiveness of this method [1]. Deep learning outperforms most other traditional machine learning techniques and has the added benefit of not requiring hand crafted features to be effective. Sirinam et al. [1] show that deep learning is an effective approach for website fingerprinting attacks against encrypted networks such as Tor, and they show that it is also effective against some defenses, such as WTF-PAD that seeks to add padding only on low usage of the channel to save bandwidth while still masking traffic bursts and the features they create. Even in an open-world setting their results show high precision. Whereas other defenses proved more effective in an open world setting. More exploration of the models, such as amount of convolutional layers, depth or dropout parameters, may yield better results [1].

Matthews et al. [15] cover nine notable, recent and efficient website fingerprinting defenses in their 2022 paper. They determined that the strongest defense without excessive overhead in a closed-world environment was TrafficSliver. However, the defense only works against certain attackers due to how it splits traffic across multiple connections. The best security provided by a non splitting defense was Interspace that cut accuracy to 76.1 percent while maintaining 98 percent bandwidth overhead and adding no additional delays. Others, such as FRONT, required more bandwidth overhead at 48 percent and no added delays yet only cut accuracy to 81.8 percent [15]. From the view of both the user and the developers it may not be worthwhile to have a 50 to 100 percent increase in overhead if an adversary can still get reasonable accuracy. The continual evolution of attacks and defenses shows that Tor security and anonymity may not be as guaranteed at any given time as a user thinks it is.

2.5 Video Fingerprinting

Leveraging the same line of thinking, video fingerprinting is able to identify patterns from network traffic of encrypted video streams to be able to predict the video that is being watched. When a web page loads most of the content is lightweight compared to a video loading, so the UI, application and possibly an ad or two load also, leaving what remains after that to be the video loading. Schuster et al. [2] describe, “the MPEG-DASH streaming video standard contains an information leak: even if the stream is encrypted, the

segmentation prescribed by the standard causes content dependent packet bursts. We show that many video streams are uniquely characterized by their burst patterns, and classifiers based on convolutional neural networks can accurately identify these patterns given very coarse network measurements.”

Video content that is streamed is usually done at a rate of one to two times the presentation speed and causes the network traffic to be transmitted in bursts. Schuster et al. [2] state “packet bursts in encrypted streams correspond to segment requests from the client and that burst sizes are highly correlated with the sizes of the underlying segments. They also show that this information leak is a fingerprint for 20 percent of YouTube videos because their burst patterns are highly distinct.” Even if the network traffic is encrypted at the transport layer, everyone watching the network can see the size and arrival times of the traffic.

Rahman et al. [3] show that video fingerprinting can be applied to streamed video content by examining traffic traces from 50 YouTube videos collected using the Tor Browser Bundle and Selenium. They used the same methods as a deep fingerprinting attack, and did not address the differences between the network traffic of a loaded web page and streamed video. They were still able to achieve 55 percent accuracy. This shows that leveraging stronger models and accommodating the different characteristics of the network traffic could yield even more promising results. Lashkari et al. [16] proved that Tor traffic can be characterized and detected solely based off time based features. They observed the traffic flows between a Tor client and a Tor entry node and proved that with only 10 features Tor traffic can be characterized to detect different applications such as VoIP, audio streaming, video streaming, file-transfer etc.

Sangwook et al. [17] show that video fingerprinting can be done on traffic from Long Term Evolution (LTE) networks. They achieved accuracy up to 98.5 percent using 46,810 traces of LTE traffic that covered three video streaming platforms from three different cellular operators. They were able to show that video fingerprinting is possible in cellular networks despite some practical problems rising from how the traffic is transmitted.

2.6 Machine Learning Methods

Goodfellow [18] describes Convolutional Neural Networks as “a class of artificial neural network that uses a mathematical operation called convolution in place of general matrix multiplication in at least one of their layers.” Convolutional neural networks are made up of three types layers: input, hidden, and output. Data is accessed from the input layer and then goes through a series of hidden layers where the data is transformed before going to the output layer, which is the prediction the network has made. Schuster et al. [2] state that “Representation of local features (spatially local in an image or temporally local in a time series) is created by layers that apply the same linear transformation on multiple windows of input data.” Similar to Schuster et al. Tor produces network traffic that creates bursts that occur in temporal proximity to each other, implying that this model should perform well. Sirinam et al. [1] explain “convolutional neural networks have become the gold standard in image classification after Krizhevsky et al. won the Large Scale Visual Recognition Challenge in 2012 and this suggests that convolutional neural networks could be useful for website and video fingerprinting attacks as well.”

2.7 Related Work

Research from Rahman et al. [3] contained a dataset of 50 YouTube videos collected from a user using the Tor network and used a model created for website fingerprinting. Research from Campuzano [5] contained only one genre of videos from YouTube with an aim to predict the video ID using the same model. Research from Kim [4] had three different video genres with eight videos each from YouTube using the same model. Research from Du’he [6] contained four genres of videos from YouTube and increased the number of videos to 35. His goal was to predict the video ID and genre using the same model created for website fingerprinting.

CHAPTER 3: Methodology

The early stages of this research were spent studying previous work in the field to include a literature review of background material about Tor, website and video fingerprinting as well as machine learning tools. Originally a dataset was collected locally over a home network to emulate the behavior and traffic of a regular user. We decided to use a more robust dataset that was collected by setting up instances of the crawler on Amazon Web Services as this new dataset will have many more instances of each video.

3.1 Data Collection

Since this work aims to compare results from previous works that collected only from YouTube, a new dataset had to be collected. The streaming platforms Vimeo, YouTube, Rumble, and Facebook video were selected as the streaming services. Our goal was to select other popular competitors to YouTube. Vimeo has been a popular streaming site since 2004, whereas Rumble is a relatively new platform created in December 2021, and Facebook video originally launched in 2017.

Other popular video streaming services such as Dailymotion, Dtube, Metacafe and Odysee were tried but we experienced technical difficulties during implementation of the crawler to stream the video and collect the network content and excluded their selection for this research. Only YouTube had an application program interface (API) that would allow us to program the crawler to automatically skip an ad and deterministically end the video then play to the next video in the list once the current video had finished playing.

We used the appearance of certain elements of the web page that indicated an ad to click on that position so that the video would play. We also added an extra 10 seconds to account for any potential buffering of the video. This resulted in most captures continuing for about 10 seconds after the video finished. Many other captures were rendered unusable for reasons such as the page element not being loaded in time for the simulated click to bypass the ad, the streaming platform taking too long to buffer the video, or fail to load based on the region Tor exit relay was in. Some platforms would load a different type of ad depending

on which region of the world the user was from, so the video would not play because the crawler simulated clicking at the wrong spot or at the wrong time. A workaround that worked for some of the video players such as Vimeo and Dtube was to load the video using the uniform resource locator (URL) of an embedded video player with an auto play feature which appeared to bypass most ads; however, some video players such as Odysee this worked locally on my machine but the video would not play on the Tor crawler. The end result was that the captures needed to be monitored closely to verify they were valid and this resulted in many unusable captures, as well as capturing data after the video had stopped playing.

Increasing the number of genres and the number of videos increased the difficulty of data collection. Finding ten videos that accurately represent the genre varied in difficulty depending on the platform. By increasing the number of media streaming platforms to four and the number of genres to six we further increased the time it took to find videos accurately representing the genre. An issue of locality arose when attempting to stream music videos, especially from Facebook. Depending on the region of the world the Tor exit relay was located many videos were not playable due to licensing issues. This led us to choose orchestra music instead. The challenges we faced in collecting this dataset will likely be similar to those an adversary will face when collecting an adequate dataset to train on.

30 virtual machines were run on Amazon web services, hosted from different regions around the world. A video ranging from three to six minutes was streamed from each, amounting to about 240 videos per day for each virtual machine. The six video genres are explained further in detail later in this paper. Screenshots were periodically taken to ensure the video was being streamed properly and logs were collected. These screenshots were manually checked to ensure that there were no errors or videos taken down. In one instance a new video had to be chosen as it was removed from the platform. Other errors were corrected when they were noticed, such as errors loading ads on Rumble and the crawler becoming stuck on a video that was not playing correctly.

The size of this dataset represents a noticeable increase compared to the next most robust dataset. Previous work from Rahman et al. [3] contains approximately 32,000 instances of 50 videos from YouTube that are around three minutes in length after they removed bad

captures. This dataset contains 213,000 instances of videos, an order of magnitude more than the dataset used by Rahman et al. This dataset will provide more training data for our model and reduce the chances that the model will overfit a small dataset that doesn't accurately portray the range of possibilities. The larger dataset should also result in increased accuracy as it is less likely to overfit the data and should help build a more generalized model by providing a wider variety of examples. The large dataset will allow the model to better capture the underlying patterns in the data which should translate to the model performing better on new data not seen before.

However, a larger dataset has more potential for outliers, noise and data that does is not relevant or representative of the dataset. Collecting the data required purchasing virtual machines through Amazon web services and taking two months to collect the data. Training the model on this larger dataset also required more computational resources such as memory and storage.

Six genres of videos were chosen with 10 videos in each genre being chosen. The genres chosen were: Kid's animated video clips such as cartoons and Disney movie scenes, sports highlight videos showing moving players, orchestra concert videos showing music, news videos of talking heads with transitions to other clips, nature videos showing slower moving transitions of wildlife, and math instructional videos showing a whiteboard or slides along with a person's voice explaining the content. The goal was for each genre to be distinct from other genres, so while each of these categories may show a person at some point the content of the video is different: the sports reel could show multiple transition clips of a baseball player whereas the news clip will mostly focus on the talking head of the anchor. Initially music videos were chosen as a genre but after discussions it was decided to use orchestra music only as the others had too much variability within and between music genres. This allowed us to have six genres distinct from each other that have videos within the genre that are similar.

To choose the videos we tried to emulate a real user as much as possible. We went to the streaming services and searched for content such as sports clips, Disney movies, nature videos, etc. and chose 10 that fell within three to six minutes in length that were a good representation of the genre. Rumble required some refining of the search query to produce appropriate math instructional videos, as most of the results were being populated with

kindergarten math. Specific queries such as differential equations and others were searched to be able to find instructional math videos similar to those of the other streaming services.

3.2 The Crawler

Collecting the dataset for analysis is arguably the most important part of this research project as it will yield a better model and more realistic results. A modified version of the Tor-browser-crawler was used [19] similar to previous work by Kim [4] and Campuzano [5]. It is a tool that creates a Tor user browsing session and then goes on to stream the videos within the video list. We assume that an adversary can access the data between the user who is the victim and their Tor entry node. This tool is written in the Python language and uses tcpdump, a packet analyzer to collect the network traffic that is streamed over a network interface. The crawler starts tcpdump with certain flags specified to create an output file and write the packets to it, the interface from which to collect the packets, and how long to collect the packets for before moving on to the next video. The crawler has a list of video URLs saved called “videos.txt.” that specifies which videos are streamed and the order that they are streamed in, once it finishes the list it begins a new run starting at the first video. During this process a few screenshots are taken per video to help detect and solve errors as well as proof that the crawler is running as it should be. A PCAP (packet capture) file is created for each video that contains the packets of each video crawl and the screenshots are saved with it. Each PCAP, along with screenshots, is stored in a different folder and each run is saved in a separate directory.

3.3 The PCAP Parser

The PCAP parser is used to extract the features contained within the packet captures. Written in python, the parser will parse each directory where the different regions of Amazon web services captures are stored. Using dpkt, the network traffic with the video stream is written to a comma separated values (CSV) file. All traffic except the connection that is the assumed to be the Tor connection is filtered out. We assume the connection with the heaviest TCP flow is the Tor connection. Features such as region, crawl ID (Identifier), video ID, genre, number of packets etc. are extracted along with the size and direction the traffic is flowing, either from the client into the Tor network or vice versa. Then these files are read into a

dataframe and used by our machine learning algorithm to find the underlying patterns that are leaked from the encrypted network traffic.

3.4 The Threat Model

The threat model used in this research assumes that the adversary has the ability to passively eavesdrop and intercept the user's encrypted Tor session before it enters the Tor network through the guard node. A nation state that controls and has access to the Internet Service Provider (ISP) and the data transferred, malicious actors gaining access to your switch or router and able to remotely monitor the flow of network packets, or by packet sniffing if the user is using a wi-fi network.

Attacks can either be focused on one specific user and his/her specific network traffic or more broadly, such as that from a guard node or an ISP. A focused attack has the ability to train a classifier off the user's network traffic and has the potential to be more successful. A broader style attack is done against many users and may be done by a nation state to screen whole populations who visit content over the Tor network that they have prohibited. Other assumptions made involve the user's browsing behavior. A browser is opened visiting one website, streaming one video and anything else that the website page loads, then moving on to do the same with the next video in the list. This may not fully reflect the average user's browsing (who may have their email open, be streaming music or have a VOIP (Voice Over Internet Protocol) program open to talk to friends). However, due to the reduced speed of streaming video over Tor compared to not using Tor, this is more likely to be accurate for a user that is using the Tor network.

In this research, data is represented as a plus or a minus along with the packet size and direction. Sirinam et al. [1] reduce traffic to a positive or negative one to indicate whether it was sent from or to the client. Hayden [20] includes the packet sizes as well as the directions. We believe keeping the packet sizes is a better representation of the network traffic than reducing the traffic to simply plus or minus one's since a cluster of Tor cells can be wrapped in TCP segments that is wrapped inside an IP packet. When reducing the traffic to one or minus one the size of the traffic is lost information that can no longer be used by the model to help predict the video or genre. Figure 3.1 shows part of a network capture for a YouTube video in which it is clear that multiple Tor cells are sent over one IP packet many times and

that by removing the size we are losing information that can be used to help classify a video.

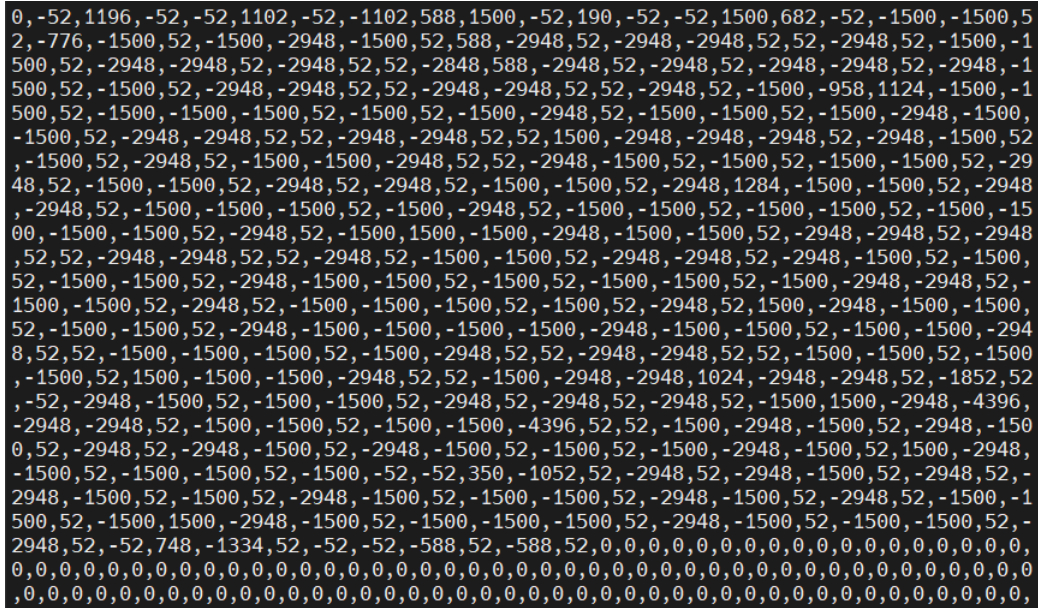


Figure 3.1. Visualization of a partial network traffic trace for a YouTube video.

3.5 Lab Environment

The first lab environment was designed to collect the network traffic transferred over the Tor network and the second lab environment was used for training the machine learning model on the collected data. For the collection environment it was originally run on a home network to better resemble a normal user. However, we decided to use the dataset collected over Amazon Web Services across various vantage points around the world due to size of the dataset. We expect this dataset to yield results that more closely resemble the real world and that are more accurate. The machine learning environment utilized the Naval Postgraduate School's supercomputer, Hamming, to conduct data processing and run the machine learning models on the collected data. This enabled us to use NVIDIA Graphics Processing Units (GPU) to accelerate processing speeds and decreased the amount of time required to train a model.

CHAPTER 4:

Implementation and Design

This chapter covers an overview of the data collection and feature selection. We discuss the thought process behind our training and testing split and the results we achieved.

4.1 Data Collection

The initial plan to collect a dataset locally from a home network created two issues. The first issue was the time required to collect the dataset. We wanted to collect the dataset from four media streaming platforms, which would require either four devices running simultaneously or fewer devices alternating the streaming source. This would have increased the time required to gather an adequate dataset. Using a laptop and an Intel NUC (Next Unit Computing) device would have taken over six months to collect a comparable dataset as we did not have the ability to run these devices constantly. The second issue is that the Naval Postgraduate School (NPS) campus does not allow Tor connections. Using Amazon web services minimized the time required to collect the dataset and solved the problem of connecting to the Tor network on campus. Moving to Amazon web services also minimized local outages and streamlined the collection process. Section 3.1 covered data collection in further detail. We believe the size and quality of this dataset will produce more realistic results.

4.2 Feature Selection

The data collected using the tor-browser-crawler was parsed and split into four separate CSV files, one for each of the four media streaming platforms: YouTube, Vimeo, Facebook, and Rumble. These CSV files were then read into dataframes separately, with the data represented as columns. These columns represented the features video ID, crawl ID, platform, number of packets, genre, entry IP address, region and direction and size of the packets.

The columns, crawl ID, platform, number of packets, entry IP address, and region are not used and are dropped from the dataframe. When predicting genre, video ID and genre are dropped from the dataframe and the genre column is used to create a new dataframe that is

the Y dataframe. When predicting video ID, video ID and genre are dropped and the Video ID column is used to create a new dataframe that is the Y dataframe. The Genres are as follows: instruction, animated, orchestra, nature, sports or news. Genres are explained in further detail in section 3.1. There are a total of four media streaming platforms, six genres per platform and ten videos per genre. The video ID is the numeric value 0-239 given to each specific video out of the 240 videos in the dataset. Using a one-to-one correlation each of the 60 videos for every platform will be mapped to a number 0-59. For example, the video IDs for YouTube are 0-9, 40-49, 80-89, 120-129, 160-169, and 200-209 which are then converted to 0-59 because the convolutional neural network model expects the values 0-59 when predicting a 60 way classifier. Crawl ID is the numeric value assigned to the crawl based on the date it ran. Platform is the streaming platform the media was played from either YouTube, Vimeo, Rumble, or Facebook. Number of packets is the amount of packets captured for the streamed video. Entry IP address is the IP address of the entry relay for the Tor network. Region is the physical location the Amazon web services virtual machine was located.

The direction and size of packets represents whether the client was sending information or receiving it and the size in bytes. We believe that including the size will allow for a better representation of the video stream, as in many cases multiple tor cells are sent at once. Rather than reduce the size of traffic to only a 1 or a -1 to represent the direction of traffic, we decided to keep the size. Traffic in our research is represented by a positive or negative number depending on the size of the packet, as shown in Figure 3.1.

4.3 Train-Test Split

Using standard supervised machine learning procedures we split our dataset into training, testing and validation sets. Following Campuzano [5], Kim [4] and Du'he's [6] research, we used a 70/30 split that resulted in 70 percent training data and 30 percent testing data. The validation set was comprised of half of the data in the test set. Splitting the data this way helps reduce overfitting while also maintaining a large training set to improve the accuracy of models.

When predicting genre we decided to manually split the data for each platform by video, using the last three videos of each genre as the testing set and the first seven as the training

set. The simple static assignment of training videos is sufficient for us to compare initial results from predicting genre to results predicting the video collection. This was done to more accurately represent the model predicting the genre of a new unseen video. A normal training and testing split resulted in instances of each video being in both the training and testing sets. After each platform's network traffic captures were read into a dataframe, the training and testing splits were conducted and then saved to be used with the machine learning model.

4.4 Comparison to Previous Work and Results

We have a total of six genres and 60 videos in our dataset, compared to previous research looking at datasets including one or four genres and including nine to 50 videos. Each dataset from the four streaming platforms has about 53,000 instances of videos and a total of 213,000 instances of videos across the four media streaming platforms. Rahman et al. [3] have a dataset of 50 YouTube music videos and about 32,000 instances of those videos after removing instances with low packet counts. They were able to achieve almost 55 percent accuracy on their dataset. Campuzano [5] achieved 84 percent accuracy when predicting video ID for one genre and nine videos of YouTube videos. Kim [4] achieved 63 percent accuracy but increased the amount of YouTube videos to 27. Du'he [6] achieved 61 percent accuracy but increased the genres to four and the amount of YouTube videos to 35 across the genres.

We expect our model to have a more difficult time predicting video ID than in previous research due to the increase in videos. The classifier is now a 60-way classifier since it has to predict the video ID out of 60 possible options. For our research we split up the four platforms and our video ID prediction accuracy is as follows: YouTube scored 78.87 percent, Vimeo scored 92.72 percent, Rumble scored 31.82 percent and Facebook scored 88.35 percent. We believe the low scores for Rumble are due to the low variety of ads present on the streaming platform combined with the long duration of the ads, and the fact that some videos play an advertisement both at the beginning and in the middle of the video. This results in similar content appearing across many different videos.

Rahman et al. [3] achieved almost 55 percent accuracy using a dataset of 50 videos from YouTube of similar length and approximately 32,000 total videos. Our dataset increased the

number of videos to 60, total instances of video captures to roughly 213,000 and included roughly 53,000 instances of YouTube videos, achieving a 78.87 percent accuracy rate, a substantial increase. Figure 4.1 shows a visual comparison of video ID prediction accuracy for Rahman et al. [3] Campuzano [5], Kim [4], Du'he [6] and our results.

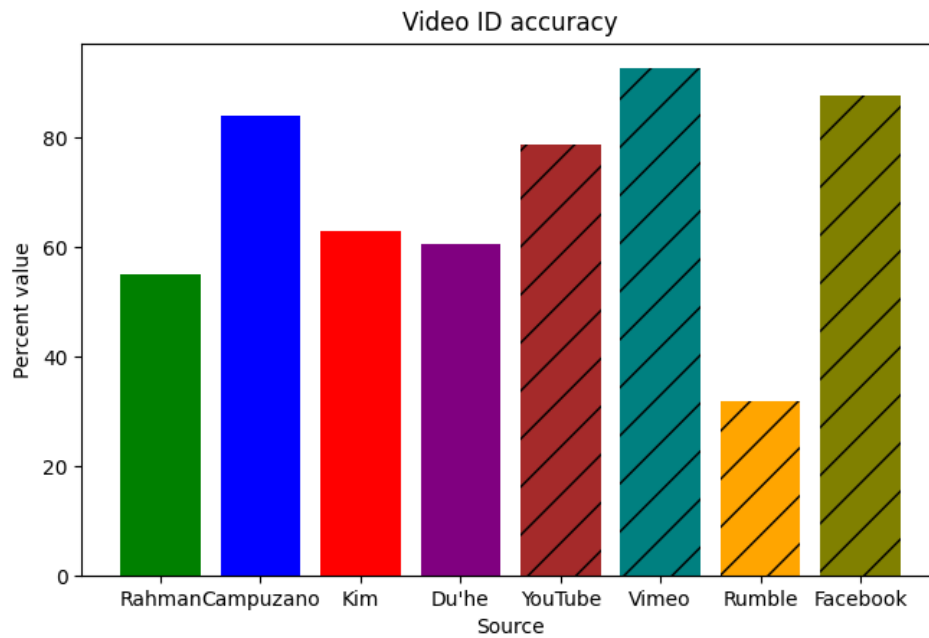


Figure 4.1. Video ID prediction accuracy for Campuzano, Kim, and Du'he. Adapted from [3]–[6]. Our research is shaded.

Previous research by Du'he [6] performed a standard training and test data split on his dataset, achieving 87.79 percent accuracy. We believe this resulted in instances of each video in both the training and testing datasets, allowing the model to learn the video played and thus making it far easier for it to predict the genre. For these results we mirrored this training and testing split and achieved accuracy as follows: YouTube scored 93.56 percent, Vimeo scored 90.82 percent, Rumble scored 51.36 percent and Facebook scored 88.35 percent. Our research does not see the correlation of reduced accuracy as the size of dataset increases that Du'he [6] mentions, and instead we see an increase in accuracy

on our YouTube data. We attribute the slightly increased accuracy scores to the increase in training data available to the model. As mentioned in the previous paragraph, we believe the abundance of similar advertisements on Rumble is the cause for the reduced accuracy. Figure 4.2 shows a visualization of the video collection prediction accuracy.

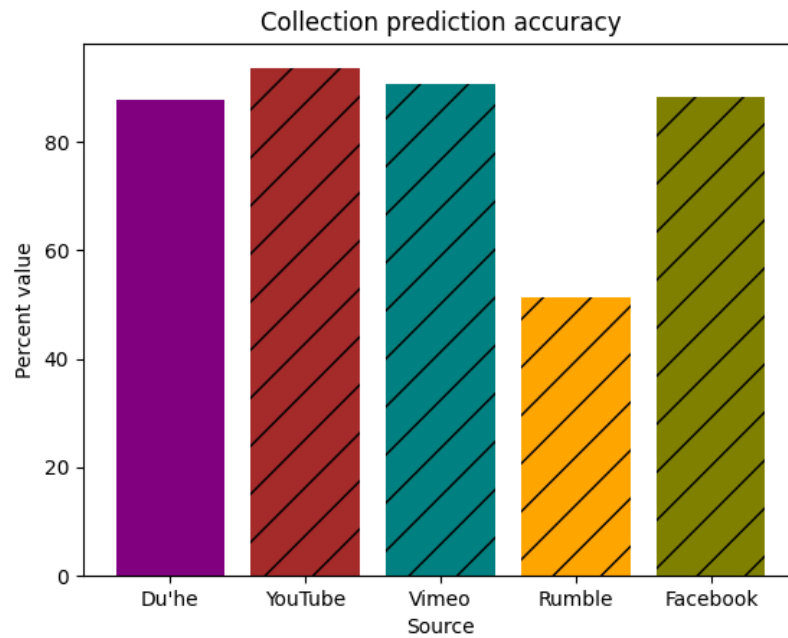


Figure 4.2. Collection prediction accuracy for Du'he and our research. Adapted from [6]. Our research is shaded.

When conducting a standard training and test data split, the same video appears in both the training and testing dataset. To accurately represent predicting genre classification on an unseen video we decided to manually split the data into training and test datasets. This was done in a simple implementation for an initial observation of the results. For all four platforms and all six genres of videos we split the last three videos into the testing set and left the first seven as the training set. For example, videos 17-19, 27-29, etc. are the testing set and videos 10-16, and 20-26, etc. are the training set. We believe this is a more accurate representation of attempting to predict the genre as the model will use the similarities in the

network traffic to predict which genre it belongs to without having examples of the video itself to compare against. The results we achieved for accuracy are as follows: YouTube scored 63.58 percent, Vimeo scored 41.86 percent, Rumble scored 40.53 percent, Facebook scored 46.93 percent. We believe this to be a more accurate representation of predicting the genre of an unknown video. Figure 4.3 shows a visual representation of these results.

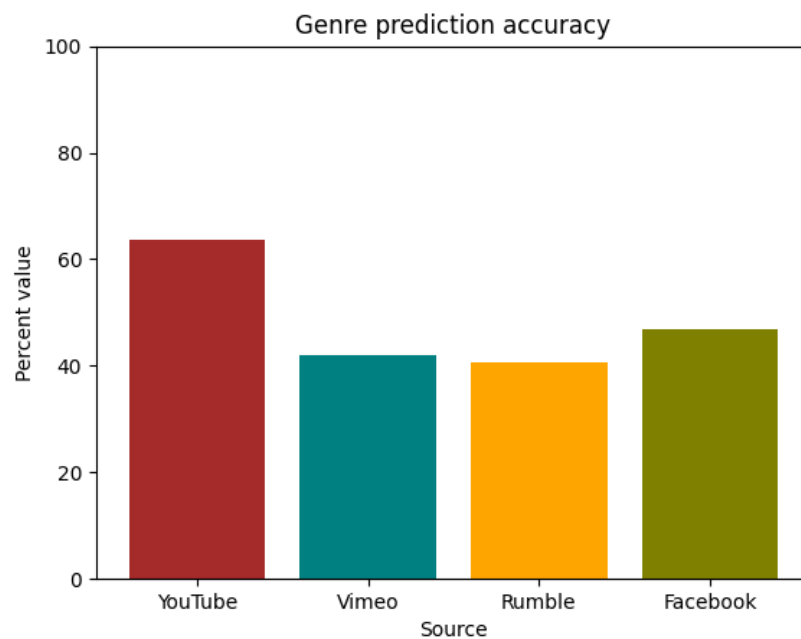
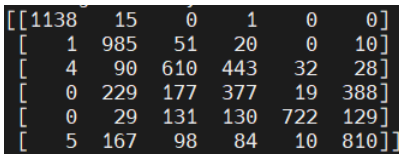


Figure 4.3. Genre prediction accuracy for YouTube, Vimeo, Rumble and Facebook when using the last videos of each category.

Figures 4.4 through 4.11 are confusion matrices that show where the model correctly predicted a genre and where it incorrectly predicted a genre. The true label is the actual label and represents the correct genre being predicted and is represented by the rows on the graph. The predicted label is the genre that is predicted by the classification model. False positives will be elements in a row that are off-diagonal. For example, the cell for genre instruction in the column animated one is 15, meaning that 15 instances from animated genre were incorrectly classified as the instruction genre. False negatives are the elements

that are off-diagonal in a column. For example, the cell for the animated genre in the instruction column is labeled one, indicating that one instance from the instruction genre was incorrectly classified as the animated genre. Diagonally, where we see the highest concentration of color, correlates to the highest correctly predicted labels, or the highest accuracy. Figures 4.4 and 4.5 show a confusion matrix for the genre prediction results on the YouTube dataset. The instruction genre has the most true positives and the least false positives while the nature genre has more false positives than it does true positives, with most being mislabeled as the orchestra genre.



A confusion matrix for YouTube Genre prediction. The matrix is a 6x6 grid of numbers. The first row (Instruction) has values [1138, 15, 0, 1, 0, 0]. The second row (Nature) has values [1, 985, 51, 20, 0, 10]. The third row (Orchestra) has values [4, 90, 610, 443, 32, 28]. The fourth row (Pop) has values [0, 229, 177, 377, 19, 388]. The fifth row (Rock) has values [0, 29, 131, 130, 722, 129]. The sixth row (Other) has values [5, 167, 98, 84, 10, 810].

[	1138	15	0	1	0	0]
[	1	985	51	20	0	10]
[	4	90	610	443	32	28]
[	0	229	177	377	19	388]
[	0	29	131	130	722	129]
[	5	167	98	84	10	810]]

Figure 4.4. Confusion matrix for YouTube Genre prediction.

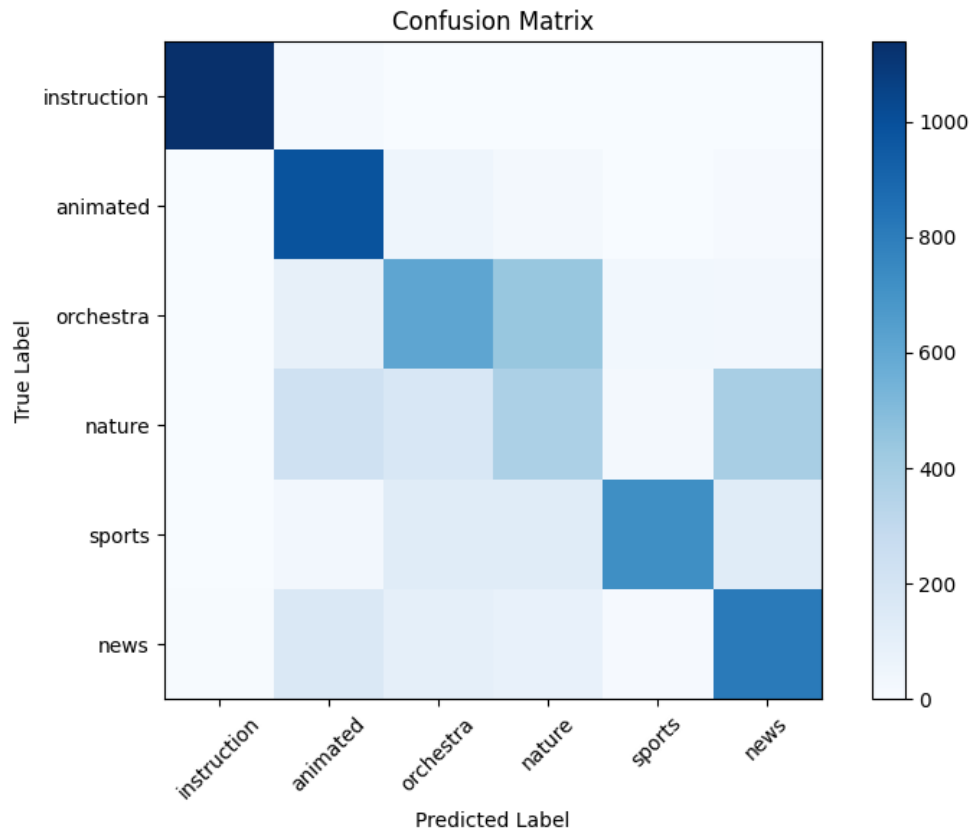


Figure 4.5. Confusion matrix for YouTube Genre prediction with color density indicating correct labels.

The confusion matrices shown in Figures 4.6 and 4.7 reveal similar predictions for Vimeo, with the instruction genre achieving the highest true positive and least false positives. The nature, sports and news genres all perform fairly poorly with many other genres being predicted incorrectly.

[	1072	8	0	0	6	2]
[	3	427	73	248	161	198]
[	47	213	315	108	182	210]
[	1	52	41	348	162	485]
[	1	5	33	327	299	381]
[	1	98	360	235	162	220]

Figure 4.6. Confusion matrix for Vimeo Genre prediction.

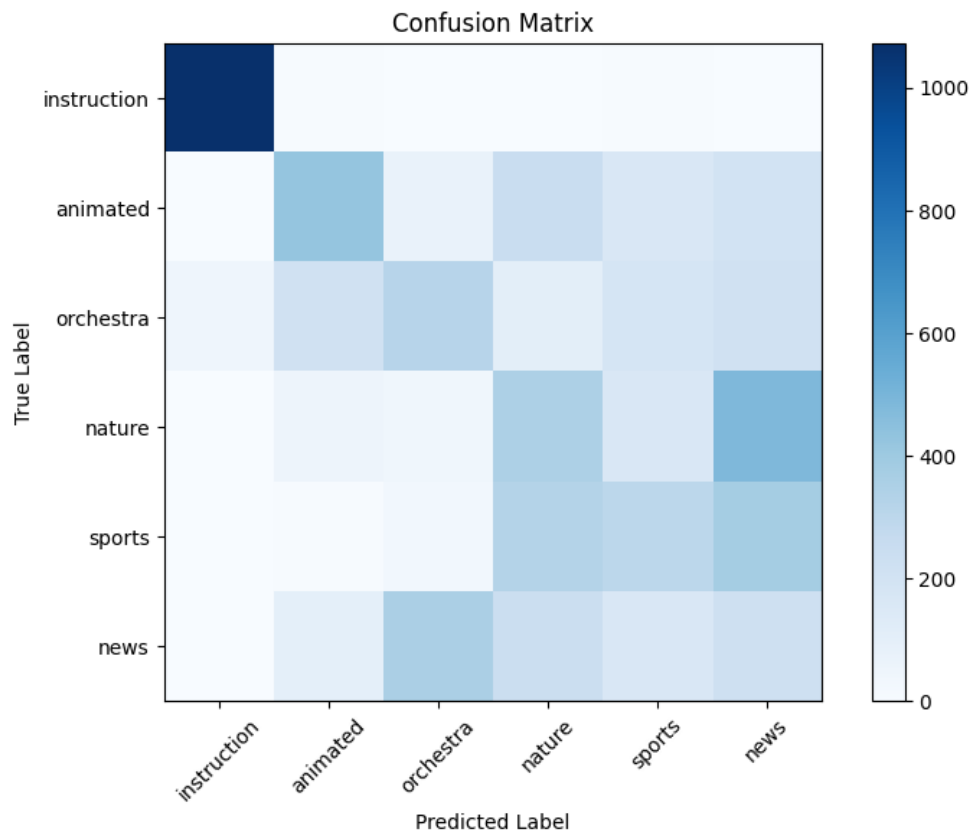


Figure 4.7. Confusion matrix for Vimeo Genre prediction with color density indicating correct labels.

Similar results are seen in Figures 4.8 and 4.9 for Rumble. The instruction genre has the highest true positive but on Rumble also has more false positives than on other platforms. The other genres all have more false positives than true positives. We expect that this is because on Rumble the advertisements are longer than the other media platforms and

sometimes multiple are played. There is also a much smaller pool of advertisements that are played. The result is that the same advertisements are played across multiple genres and multiple videos. We believe that model has a more difficult time learning the underlying pattern of each genre because there are more similarities due to the advertisements.

[	1040	16	25	9	2	5]
[	663	249	55	36	23	40]
[	200	271	269	152	126	18]
[	32	249	127	403	134	131]
[	69	133	386	255	176	28]
[	55	77	170	49	179	470]

Figure 4.8. Confusion matrix for Rumble Genre prediction.

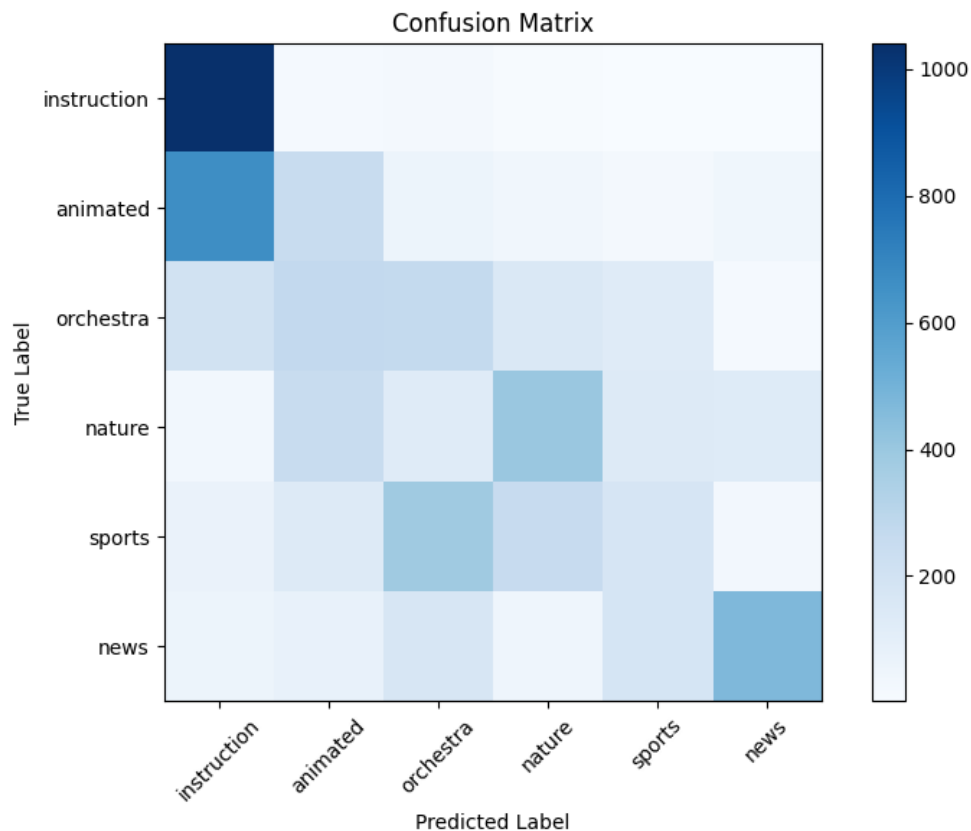


Figure 4.9. Confusion matrix for Rumble Genre prediction with color density indicating correct labels.

Figures 4.10 and 4.11 show the confusion matrices for Facebook. The instruction genre has the highest true positive but the other genres all have more false positives than true positives.

[	1059	13	21	3	4	4]
[	0	282	331	24	25	450]
[	0	98	218	187	332	229]
[	0	7	76	498	213	272]
[	0	15	14	303	720	59]
[	0	15	185	280	363	353]]

Figure 4.10. Confusion matrix for Facebook Genre prediction.

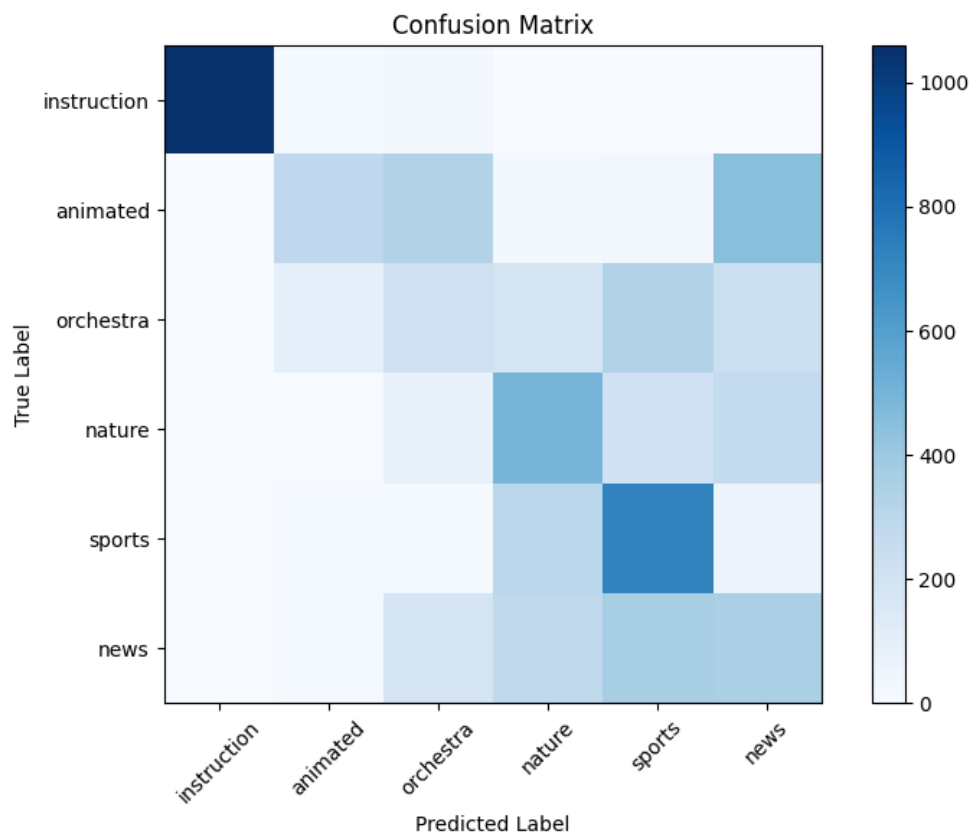


Figure 4.11. Confusion matrix for Facebook Genre prediction with color density indicating correct labels.

Each video should be a good representation of the genre it belongs to and if a video is not a good representation of the genre while being a part of the test set, we would expect to see worse accuracy scores. To compare we tested the first three videos in each genre as the testing set and the last seven as the training set. Our results for accuracy score are as follows: YouTube scored 50.06 percent, Vimeo scored 32.52 percent, Rumble scored 29.27 percent and Facebook scored 45.83 percent. Figure 4.12 shows a visual representation of these results.

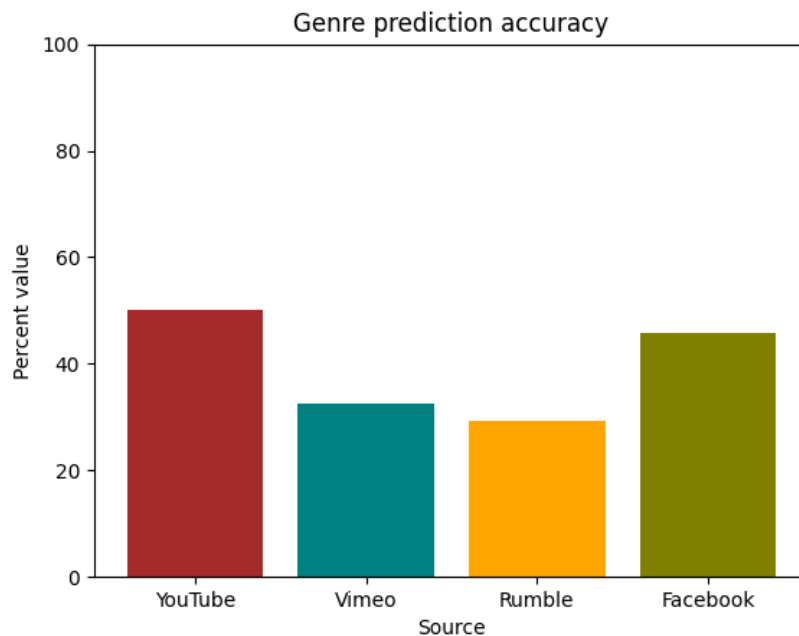


Figure 4.12. Genre prediction accuracy for YouTube, Vimeo, Rumble and Facebook when using the first three videos of each category

Figures 4.13 and 4.14 show the confusion matrices for YouTube using the first three videos as the test set. The instruction genre still performed well, yielding low amounts of false positives and false negatives. The sports genre also scored many true positives but 911 instances of the orchestra genre are predicted as the sports genre. Similarly, 799 instances of the news genre are predicted as the nature genre. From these comparisons it is clear the

the instructional genre is the only genre that is unique enough to consistently score high true positives, low false positives and low false negatives.

[	[	1076	9	1	3	0	7]
[	1	471	61	212	361	52]	
[	4	8	200	28	911	14]	
[	8	166	19	547	269	139]	
[	0	1	1	49	1061	58]	
[	0	36	15	799	159	158]]]	

Figure 4.13. Confusion matrix for Facebook Genre prediction.

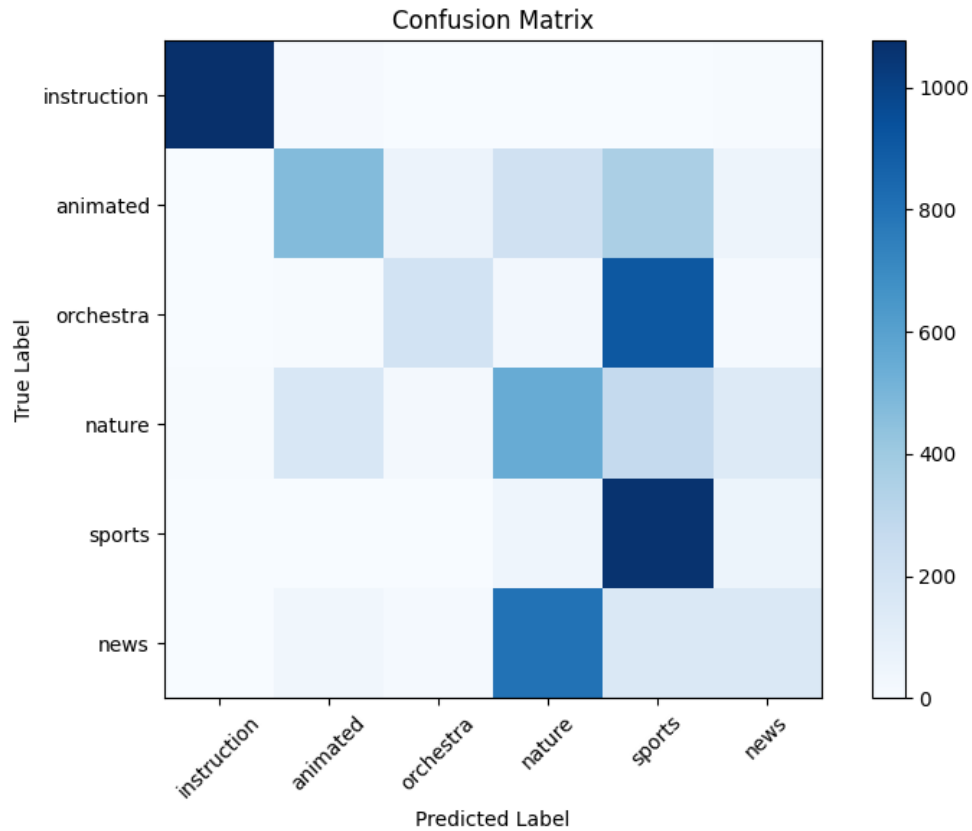


Figure 4.14. Confusion matrix for YouTube Genre prediction using first three videos as test set with color density indicating correct labels.

We believe the instruction genre to score high accuracy across platforms because it is a genre of videos recorded by a teacher and uploaded to each platform. All the videos were likely recorded with the same device and same settings, and in a similar physical environment. Other genres such as sports include multiple sports reels, and hockey highlights are going to look different from another sport such as baseball highlights. This can similarly be applied across the other genres, there is larger variety within the genre than within the instruction genre. A more accurate prediction could likely be obtained by attempting to predict a channel on a media streaming platform, because the videos that they host are going to be more similar to each other.

We do not believe the manual 70/30 training and test split that we conducted to be the most optimal method and it was done as a simple method to observe results. Other methods such as random selection of seven videos could yield better results. We do not believe that our dataset is optimized to predict the genre of video that is being streamed. Our dataset includes many instances of seven videos for our training set and many instances of three videos all of the same genre for the testing set. Ten videos is not an accurate representation of a genre and this explains why choosing a different subset of videos for the training and testing set makes a big difference in prediction accuracy. A dataset that aims to predict genre should have many separate videos with only one instance of that video for each genre.

CHAPTER 5:

Future Work and Conclusion

This research expanded upon previous work about on fingerprinting attacks over encrypted networks such as Tor using deep learning models. Our research measured the effectiveness of the fingerprinting attack proposed by Sirinam et al. [1] in a closed-world environment against streaming platforms other than just Youtube. This attack utilizes a convolutional neural network to determine which video a user may be streaming over Tor, based on features selected from the encrypted network traffic captures.

5.1 Future Work

Future work involving an open-world scenario should provide a more realistic environment to experiment with video fingerprinting attacks. A real world scenario should include other traffic than just the video being streamed. A regular user will likely do some other form of web browsing like checking for email updates or talking to friends with voice over IP. Another problem is the attacker will not have the availability to download all of the videos that the user could possibly watch. An open-world scenario should also aim to include captures that simulate a real world network and real world scenarios, such as videos failing to buffer partially, videos that update their streaming quality due to network conditions, and users that run ad blockers for their browsers.

This research uses the model proposed by Sirinam et al. [1] and was created with website fingerprinting in mind. Fine tuning the hyperparameters to be better suited for longer network traffic captures involved in video fingerprinting is an area for improvement.

This dataset is not all inclusive. There are a wide variety of genres of videos and many other media streaming platforms that are not included in this dataset. More can be also done to emulate the different types of users and network conditions in order to achieve a better holistic view of what features make encrypted video network traffic more susceptible to video fingerprinting.

5.2 Conclusion

We measured the effectiveness of the fingerprinting attack proposed by Sirinam et al. against other media streaming platforms than just Youtube. We tested the attack against YouTube, Vimeo, Rumble and Facebook video. We found that when looking at the all of the videos as a collection or playlist, most platforms were susceptible to fingerprinting.

When predicting the video ID, Vimeo produced noticeably higher accuracy, indicating it is more susceptible to fingerprinting attacks, followed by Facebook, YouTube, and Rumble. Vimeo is a platform that requires a customer to pay to host their content and does not show advertisements to viewers. We believe this is the cause of the increased accuracy when fingerprinting Vimeo and would also explain why Rumble scored as low as it did. Rumble begins each video with an advertisement and will frequently play another in the middle of a video. Rumble also has a far less diverse pool of advertisements that it plays, so the same long advertisements will appear in multiple different videos. We believe the same advertisement appearing in multiple videos and multiple genres makes it more difficult to classify which video ID or genre is being streamed.

When predicting the collection of videos, the model produces good accuracy results for all platforms with the exception of Rumble. We believe that this is due to the aforementioned limited pool of advertisements that are played repeatedly on Rumble. The same advertisements are played across all genres, which reduces the accuracy. When predicting the genre of the collection of videos, the training and test split is done automatically. It is likely that examples of each video appear in the testing data, which will cause the model to have an easier time determining which genre that video belongs in since the model already knows that answer. This is the genre prediction done in previous work by Campuzano [5], Kim [4], and Du'he [6].

We have shown that it is possible to fingerprint undefended network traffic over Tor with high accuracy to identify the video that a user is streaming. We show that it is possible to predict the genre of a video that someone is watching but not reliably so with the current dataset. The dataset should have many examples of one instance of a video within the genre, rather than many instances of the same video that we have in our dataset.

List of References

- [1] P. Sirinam, M. Imani, M. Juarez, and M. Wright, “Deep fingerprinting: Undermining website fingerprinting defenses with deep learning,” in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018. Available: <https://dl.acm.org/doi/abs/10.1145/3243734.3243768>
- [2] R. Schuster, V. Shmatikov, and E. Tromer, “Beauty and the burst: Remote identification of encrypted video streams,” in *USENIX Security Symposium*, 2017. Available: <https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/schuster>
- [3] M. S. Rahman, N. Matthews, and M. Wright, “Poster: Video fingerprinting in Tor,” in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019. Available: <https://dl.acm.org/doi/abs/10.1145/3319535.3363273>
- [4] E. Kim, “Predicting the unknown: Machine learning techniques for video fingerprinting attacks over tor,” M.S. thesis, Dept. of Comp. Sci., NPS, Monterey, CA, USA, 2021. Available: <https://apps.dtic.mil/sti/citations/trecms/AD1164933>
- [5] C. Campuzano, “Towards video fingerprinting attacks over tor,” M.S. thesis, Dept. of Comp. Sci., NPS, Monterey, CA, USA, 2021. Available: <https://apps.dtic.mil/sti/citations/trecms/AD1164219>
- [6] P. Du’he, “Removing the mask: Video fingerprinting attacks over tor,” M.S. thesis, Dept. of Comp. Sci., NPS, Monterey, CA, USA, 2021. Available: <https://calhoun.nps.edu/handle/10945/71974>
- [7] P. Biddle, P. England, M. Peinado, and B. Willman, *The Darknet and the Future of Content Protection*. Berlin, Germany: Springer, 2003. Available: http://dx.doi.org/10.1007/978-3-540-44993-5_10
- [8] R. Gehl, *Weaving the Dark Web: Legitimacy on Freenet, Tor, and I2P*. Cambridge, Massachusetts: MIT Press, 2018. Available: <https://doi.org/10.7551/mitpress/11266.001.0001>
- [9] S. Samtani, H. Zhu, and H. Chen, “Proactively identifying emerging hacker threats from the dark web,” *ACM Privacy and Security*, vol. 23, no. 21, pp. 2–33, Aug. 2020. Available: <https://dl.acm.org/doi/10.1145/3409289>

- [10] The Tor Project, “Users - Tor metrics.” Accessed: Jul. 15, 2023. Available: <https://metrics.torproject.org/userstats-relay-country.html?start=2020-04-01&end=2023-04-10&country=all&events=off>
- [11] R. Dingledine, N. Mathewson, and P. Syverson, “Tor: The second-generation Onion Router,” in *Usenix Security*, 2004. Available: https://www.usenix.org/legacy/event/sec04/tech/full_papers/dingledine/dingledine_html/
- [12] The Tor Project, “Types of relays,” Accessed: Jul. 15, 2023. Available: <https://community.torproject.org/relay/types-of-relays/>
- [13] L. Goldberg, “Tor: Anonymity for better or for worse,” *Systems and Network Security*, accessed July 18, 2023. Available: <https://medium.com/systems-and-network-security/tor-anonymity-for-better-or-for-worse-d8407b1d9287>
- [14] A. Hintz, “Fingerprinting websites using traffic analysis,” in *Privacy Enhancing Technologies: Second International Workshop*, 2003. Available: <https://dl.acm.org/doi/10.5555/1765299.1765312>
- [15] N. Mathews, J. K. Holland, S. E. Oh, M. S. Rahman, N. Hopper, and M. Wright, “SoK: A critical evaluation of efficient website fingerprinting defenses,” in *2023 IEEE Symposium on Security and Privacy (SP)*, 2022. Available: <https://ieeexplore.ieee.org/abstract/document/10179289>
- [16] A. H. Lashkari, G. Draper-Gil, M. S. I. Mamun, and A. A. Ghorbani, “Characterization of Tor traffic using time based features.” in *Proceedings of the 3rd International Conference on Information Systems Security and Privacy - Volume 1: ICISSP*, 2017. Available: <https://www.scitepress.org/Papers/2017/61056/>
- [17] B. Sangwook, S. Micheol, K. Dongkwan, P. Cheoljun, L. Jiho, S. Sooel, and Y. Kim, “Watching the watchers: Practical video identification attack in LTE networks,” in *Proceedings of the 31st USENIX Security Symposium*, 2022. Available: <https://www.usenix.org/system/files/sec22-bae.pdf>
- [18] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, 1st ed. Cambridge, Mass, USA: The MIT Press, 2016.
- [19] Walsh, Timothy and Matthews, Nate and Campuzano, Daniel, “Tor-browser-crawler-video,” Accessed: Jul. 15, 2023. Available: <https://github.com/timwalsh300/tor-browser-crawler-video>
- [20] B. Hayden, “Defending against deep learning-based video fingerprinting attacks with adversarial examples,” M.S. thesis, Dept. of Comp. Sci., NPS, Monterey, CA, USA, 2022. Available: <https://apps.dtic.mil/sti/citations/AD1184913>

Initial Distribution List

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California



DUDLEY KNOX LIBRARY

NAVAL POSTGRADUATE SCHOOL

WWW.NPS.EDU

WHERE SCIENCE MEETS THE ART OF WARFARE