REPUBLIC OF TURKEY

ALTINBAS UNIVERSITY

Institute of Graduate Studies

Electrical and Computer Engineering

A NOVEL STUDY OF PREVENTING THE CYBER SECURITY THREATS AND RANSOMWARE ATTACKS USING MACHINE LEARNING

Mustafa Hasan Hussein ALTAMEEMI

Master's thesis

Supervisor

Asst. Prof. Dr. Abdullahi Abdu IBRAHIM

Istanbul, 2022

A NOVEL STUDY OF PREVENTING THE CYBER SECURITY THREATS AND RANSOMWARE ATTACKS USING MACHINE LEARNING

Mustafa Hasan Hussein ALTAMEEMI

Electrical and Computer Engineering

Master’s thesis

ALTINBAŞ UNIVERSITY

2022

The thesis titled A NOVEL STUDY OF PREVENTING THE CYBER SECURITY THREATS AND RANSOMWARE ATTACKS USING MACHINE LEARNING prepared by MUSTAFA HASAN HUSSEIN ALTAMEEMI and submitted on 10/4/2022 has been **accepted by majority of votes for t**he degree of Master of Science in Electrical and Computer Engineering.

Asst. Prof. Dr. Abdullahi Abdu IBRAHIM

Supervisor

Thesis Defense Jury Members:

Asst. Prof. Dr. Abdullahi Abdu IBRAHIM	Faculty Of Engineering and Architecture, Altinbas University	_____
Asst. Prof. Dr. Sefer KURNAZ	School Of Engineering and Architecture, University	_____
Assoc. Prof. Dr. Mohammed ABUBAKAR	Faculty Of Business, Antalya Bilim University	_____

I hereby declare that this thesis meets all format and submission requirements of a Master's thesis.

Submission date of the thesis to the Graduate Education Institute: ___/___/___

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Mustafa Hasan Hussein ALTAMEEMI

Signature

DEDICATION

I devote and pledge this research work to my supervisor who is salient for guiding me through whole research work as well as my family for always assisting me in my hard time.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my supervisor Dr. Abdullahi Abdu IBRAHIM for guiding and helping me along the way in writing this dissertation. Discussing my progress, problems, and ideas with my supervisor Dr. Abdullahi Abdu IBRAHIM a couple of times every week helped me tremendously in understanding the logic behind the research. It made me better realize the technical need for this research work.

ABSTRACT

A NOVEL STUDY OF PREVENTING THE CYBER SECURITY THREATS AND RANSOMWARE ATTACKS USING MACHINE LEARNING

ALTAMEEMI, Mustafa Hasan Hussein

M.Sc., Electrical and Computer Engineering, Altınbaş University,

Supervisor: Asst. Prof. Dr. Abdullahi Abdu IBRAHIM

Date: 08 /2022

Pages: 63

The purpose of this master's research thesis is to categorize the cutting-edge AI-based cyber security network to reduce cybercrimes globally and provide a resilient cyberattack-free environment for consumers. One of the most often used real applications in areas like cost control, traffic scene analysis, and thought attack identification is attacks restriction. Contextual information, which in our study is described as the relationship between the cyber security and the Rivest-Shamir-Adleman, is important to get total comprehension along with the information on the owner vehicle. The research study suggested a KNN classifier for continuously limiting the position of organizations and digital protection in the area. The modified districts are passed to the Rivest-Shamir-Adleman after being preprocessed to reduce commotion (RSA). The preprocessing of the RSA separates associated components, divides them into lines of attack, and channels them by size. The well-known technique used in the research is RSA. The user is advised to acquire a new attack of that portion of the assault if the RSA reported a low confidence score for at least one attack region. The new attack is then compared to the existing attack, and the attack processing steps are once more carried out. If all locations have a sufficiently high score or enough retries have been used up, the security cycle ends. The outcome includes the type of assault that was discovered, a list of the districts that were subject to the removed assault, and the accuracy of the RSA calculation through KNN, which was calculated across 300 preparation and testing iterations. The proposed framework was created using 75% of

the available data, 20% of which was used for testing, and 5% for validation. We used a Cyber Data Dataset made up of multiple classes that addressed a portion of all current forms of digital protection to illustrate the meaning of the proposed component vector. Multiple machine learning toolboxes were utilized for the research effort, which was implemented and carried out in the MATLAB programming language. Our strategy has three practical head networks, so a potential preparation risk is prompted by the start to finish and synchronous preparation. It is incredibly challenging to determine which head configuration is producing a problem on the off chance that the model doesn't come together. As a result, in our plan cycle, we gradually added each helpful head to make sure the single head plan idea was functioning correctly before extending the model with the remaining useful heads.

KEYWORDS: Detection, Artificial Intelligence, KNN, Cyber Security, Feature Extraction.

TABLE OF CONTENTS

Pages

LIST OF FIGURES

Pages

LIST OF TABLES

Pages

LIST OF ABBREVIATIONS

AUC	:	Area Under Consideration
CSER		Category-Specific Extremal Regions
CAD	:	Computed Aided Design
DCNN	:	Deep Convolutional Neural Network
KNN	:	K-Nearest Neighbor
LBP	:	Local Binary Patterns
MSER	:	Maximally Stable Extremal Regions
RNN	:	Recurrent Neural Network
ROI	:	Region of Interest
RSA	:	Rivest–Shamir–Adleman
SVM	:	Support Vector Machine

1. INTRODUCTION

Electric electricity is vital to our contemporary society. As a result, a vast global network of transformers, power pylons, and other electrical power supplying equipment is dispersed for network security. Electricity must always be available; hence the underlying infrastructure must have routine maintenance as indicated in [1]. As a result, information from associated cyber security is crucial for both documentation and other maintenance procedures. According to [2], the content on these cyber security devices typically comprises of textual data such serial numbers, manufacture years, vendor names, schematic drawings, trademarks, and even barcodes. Cybersecurity is often rectangular in shape and constructed of synthetic or metallic materials. Since they come from many vendors, these attacks vary in size, layout, text, and material. The first sample cyber-attack is made of tarnished metal, has a barcode among the human readable assault, and is composed of the attack. The attacks, as indicated in [3], are composed of synthetic material, have a polished and reflecting surface, contain schematic drawings, and are a great deal bigger than the initial attack. The operator has been manually transcribing the cyber-attack data while doing maintenance activities up until this point. This transcription is vulnerable to inaccuracies, including typographical mistakes and the absence of crucial information, which could seriously harm the affected device or possibly result in as mentioned in [4]. Due to the complexity of most cyber security, this process also takes a lot of time. Recent production samples are used as demonstrations of cyber security. They are therefore brand new, have never been attached to a device, and are in perfect shape. Metal cyber security, on the other hand, that has been installed on a gadget for several decades. These examples show a few of the challenges manual transcriptions and automatic content extraction, as indicated in [5], face. The cyber security is exposed to environmental factors for many years, which causes it to age and become tarnished. It is also conceivable for very old cyber security to lose its paint, making the attack nearly unintelligible. Strong shadows, reflections, variations in illumination, and partial occlusions are further challenges that are related to the outdoor site and may be problematic when taking samples for automatic processing. The attack, which has a specular reflection at the top as indicated in [6].

Figure 1.1: The CIA tried within cyber security system with important aspects [6].

One could convert attack samples into machine-readable attack detection in order to extract the semantic information that is available on the cyber-attack. Numerous strategies have previously been developed with the goal of completing this task, in addition to the easily accessible RSA software implementations stated in [7]. However, the majority of the currently available systems are built and programmed to handle scanned paper sheets. Due to the fact that each of the four attacks in the proposed cyber security system uses a different typeface from the ones employed in attack processing as described in [8], this presents additional challenge. In addition, one assault is embossed, which means that it has a narrower stroke and less contrast than painted or printed attacks.

To identify, categorize, and extract attacks from cyber security, this research suggests a revolutionary method. With the requirement that the cyber security be rectangular and contain human readable attack, more precisely, cyber security that are attached to electric power supply equipment. Barcodes and schematic drawings are not processed. The following steps make up our strategy: The first step is to identify the cyber-attack in the input assault, extract it, and twist it so that it is upright. Then, using a machine learning strategy, we extract features and establish the kind. A list of the titles and positions indicated in [9] is provided for each kind. Each area undergoes preprocessing to get rid of noise and clutter brought on by dirt, fragments of nearby attacks, or frames surrounding the attack. The filtered regions are finally, An RSA engine receives the filtered regions as input. The user is instructed to acquire a new attack of the attack region where the issue occurred if the recognition score

of one or more regions is below a threshold.

1.1.MOTIVATION

IT provides an overview of a few relevant fields of study. The analysis of numerous comparable assignments, in particular the recognition and attack extraction from cyber security's and cyber security, as well as the location of items in checked examples and the order of structures, is done because no other calculations have been suggested that precisely match our objective. Typically, these calculations consist of identifying articles that seem to be an assault, followed by the extraction of information that appears to be an understandable attack, like the one we are concentrating on. The current attack is matched to the new one, which might only delay a portion of the attack, and the RSA is carried out once more. When obstructions or reflections are present, this multi-view technique ensures robust message extraction. The final outcome of the computation includes the distinct assault type, the perceived attack, its context with respect to the assault, and its certainty, as noted in [10].

Figure 1.2: Overview of cybersecurity stack with main fundamentals [10].

Making a model of an AI application that enables the client to receive tests on which the most recent recorded steps to extract the substance are carried out is the goal of the practical

assignment. The evaluation's primary focus is on attack categorization, where we test the elements that were collected from the assault both separately and in combination. Additionally, we demonstrate the effects of tailoring the RSA method for the assault's consumer network and the viability of the ANN application for three different attack types. Finally, we demonstrate how varied illumination conditions affect the stages of attack classification and extraction. This comprises several severe situations that end in failure.

1.2.PROBLEM STATEMENT

Our challenge of cyberattack detection and categorization share many characteristics with the task of isolating cyberattack from cyber security. The demand for vehicle security is high today and includes human readable attack. Since the Rivest-Shamir-Adleman (RSA) algorithm is used to attack each vehicle’s cyberspace, it is important to locate these attacks in order to retrieve their contents. According to [11], the researchers suggest a mechanism to identify, and monitor cyberattacks for traffic management. To find the vehicle attacks in the input data, they employ Maximally Stable Extremal Regions (MSER). While dark regions with a bright barrier are utilized to detect the cyberspace on the one hand, the suggested technique uses bright regions surrounded by a dark boundary to detect cyber-attacks This study displays the cyber-attack output of three representative cyber security systems. If several cyberattacks are discovered inside a single, more significant cyberattack, the area is identified as a cyber-security. The black areas must also roughly all have the same size and have straight lines connecting their center points. According to [12], the average height of the dark sections must likewise be like the height of the bright region that surrounds them. The research used K-Nearest Neighbors (KNN) with a one-vs-one technique to conduct character recognition, with each individual cyberspace represented by each cyberattack serving as the classifier's input. Multiple samples of the attack are available because they follow the cyber security over several frames, which are applied to raise classification accuracy. The best individual character results from a majority vote over all frames constitute the ultimate detection result of the cyber security attack.

1.3. RESEARCH CONTRIBUTIONS

The contributions to this dissertation's research are made by:

i. The goal of the study is to suggest a novel method for identifying and detecting cyber security in natural samples. However, study employs a superset of cyberattacks called extremal areas to find the attacks.

ii. It should be mentioned that this method finds attacks in samples that are natural. Therefore, a separate grouping and filtering step using the RSA algorithm must be used for each type of attack (DoS Attack, Fuzzy Attack, Spoofing the Drive Gear, Spoofing the RPM gauge, and Attack-Free (normal)) in order to obtain only consumer vehicle attack samples..

iii. The RSA technique was selected for this study because it is also gradually computable for the purpose of detecting cyberattacks, lowering the complexity of the runtime.

iv. The study suggests using normalized central algebraic moments, compactness, Euler number, entropy of the cumulative histogram, number of convexities, and area of the convex hull. Euler number is calculated by subtracting the number of attacks from the number of holes in the attacks (although the last two descriptors are not incrementally computable).

v. To list all extreme assaults at each low, medium, and high level, the suggested machine learning-based KNN technique is combined with the RSA algorithm..

vi. Intensity level and vehicle connected component attributes are extracted for each attack to reflect the most extreme attacks, and RSA algorithm descriptors are then derived for each attack.

vii. The KNN descriptors based on machine learning input each intensity level of the attack into a classifier, whose output shows whether the cyber-attack is of interest. This means that for a cyberattack to be recognized, it must be rotated or slanted and included in the training samples.

viii. In this study, the KNN cyber-attack detection pipeline and detection outcomes are provided. Consumer cyber security is improved alongside the cyberspace of the country-of-origin sticker above the consumer cyber security. Additionally, this method can be used to identify internet browsing and is not just restricted to networks.

1.4. THESIS ORGANIZATION

The commonality among the machine learning approaches discussed in Chapter 1's introduction is that they all identify potential cyber-attacks and, with the exception of the scanned cyber-attack section, extract information from consumer networks that is then either stored or forwarded for additional processing. The detection steps of these methods are discussed in Chapter 2 of the literature review. Either the cyber-attack is detected and the basic structure is exploited to group them together to find the cyber-attack, or additional constraints such a uniform background or foreground color must be met. The study technique is provided in chapter 3 with the example of processing forms; no KNN cyber-attack detection is required; only the type of the form needs to be classified using the RSA algorithm. Additionally, the Rivest-Shamir-Adleman (RSA) algorithm and content extraction are made simpler by the fact that the attack's RSA method format is often known and that only a small number of typefaces and fonts are employed. With one important exception, the experimental results are reported in chapter 4 for cyber security recognition, where a variety of fonts may be utilized. However, this also affects the examined methods negatively. Forms are frequently filled out by hand as well, but since improving cyber security is a separate topic, such instances are not covered in the research under consideration. Since these approaches exploit various specific properties of the attacks for which they are designed—properties that are not always valid in our case—none of these entire approaches can be employed as-is, a comparison with the body of current literature is offered in chapter 5. The conclusion of the proposed research method to identify and categorize cyber security attacks, utilizing the particular steps of some of the examined methods, was presented in chapter 6, along with suggestions for the future.

2. RELATED WORK

This research project is motivated by the desire to overcome the challenges and issues that cyber security identification units in KNN face due to their construction and the natural opposition that surrounds them, i.e., by creating a superior climate arrangement, a solution that will avoid the shortcomings of the conventional organizations for cyber security location and controlling tasks. In the sections below, a detailed examination of the many challenges and requirements is done from top to bottom to help explain the inspiration for this theory work. Additionally, a thorough examination of the KNN architecture and its advantages is done to help this postulation effort survive.

2.1. BACKGROUND

The process of identifying and extracting content from cyber security systems has many similarities to our effort of identifying and categorizing cyber-attacks. Vehicle security features a rectangular design with assault detection that can be read by humans. Therefore, it is important to locate them in the attack and apply a Rivest-Shamir-Adleman (RSA) to the individual cyberspace in order to extract their contents.

A system to detect and track cyber security incidents for traffic management is suggested by researchers in [13]. To identify the input attack's cyber security, they employ Maximally Stable Extremal Regions (MSER). Dark regions with a bright boundary (MSER-) are used to detect the cyberspace on the attack, while bright regions with a dark boundary (MSER+) are used to detect the cyber security. Three sample cyber security KNN outputs are shown. If several MSER- regions are located inside a larger MSER+ zone, the area is identified as a cyber-security region. Additionally, as noted in [14], the black sections must roughly all have the same size and have straight lines connecting their center points. The height of the light zone immediately surrounding the dark parts must likewise roughly match their average height. Character recognition is carried out by the authors using K-Nearest Neighbors (KNN) and a one-to-one approach, with the MSER-regions specified in [15] serving as the classifier's inputs. Multiple examples of the attack are available because they also monitor the cyber security across a number of frames.

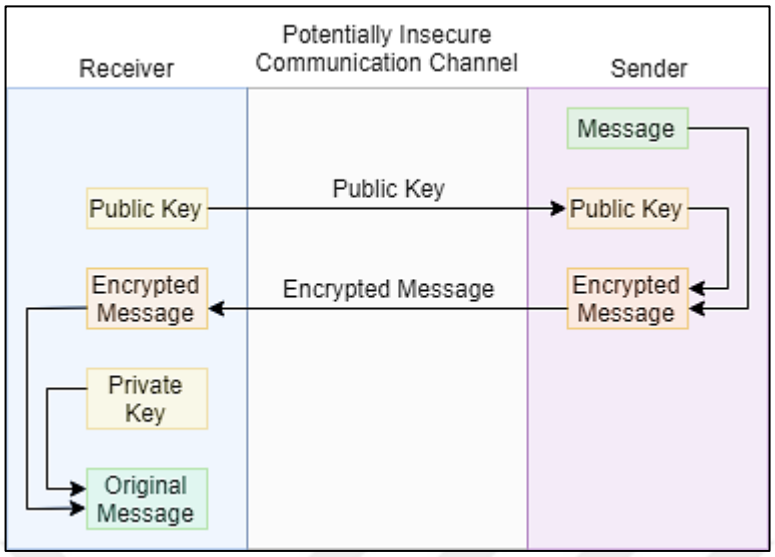

Figure 2.1: Potentially insecure communication channel for limiting the number of attacks [15].

A similar method is put forth by researchers in [16] to identify attacks in their natural positions. However, they define Category-Specific Extremal Regions using Extremal Regions, a superset of MSER (CSER). By thresholding the attack at each gray level, starting at level one as indicated in [17], the suggested technique identifies all extremal locations. Connected components that reflect the extremal regions are removed for each intensity level, and descriptors are computed. According to the research, there are only three ways that the current extremal regions can alter: they can expand, two previously independent ones can merge, or a new zone can form as described in [18]. By selecting descriptors that can also be computed gradually, this incremental updating behavior is taken advantage of, keeping the runtime complexity low. In addition to the descriptors listed in [19], they suggest using normalized central algebraic moments, compactness, Euler number (the number of holes in the objects divided by the number of objects), entropy of the cumulative histogram, number of convexities, and the area of the convex hull. The classifier that determines whether or not the region is of interest receives these descriptions as input at each intensity level. This implies that in order to detect rotated or slanted cyberspace as described in [20], it must be included in the training samples. It should be mentioned that this method finds attacks in samples that are natural. Consequently, a distinct grouping and filtering procedure must be used to only gain cyber security as mentioned in [21]. The KNN detection process and an illustration of the outcome. In addition to the cyber security, the country-of-origin sticker's cyberspace is also detected above the cyber security. Additionally, this method can be used to identify additional items in addition to attacks.

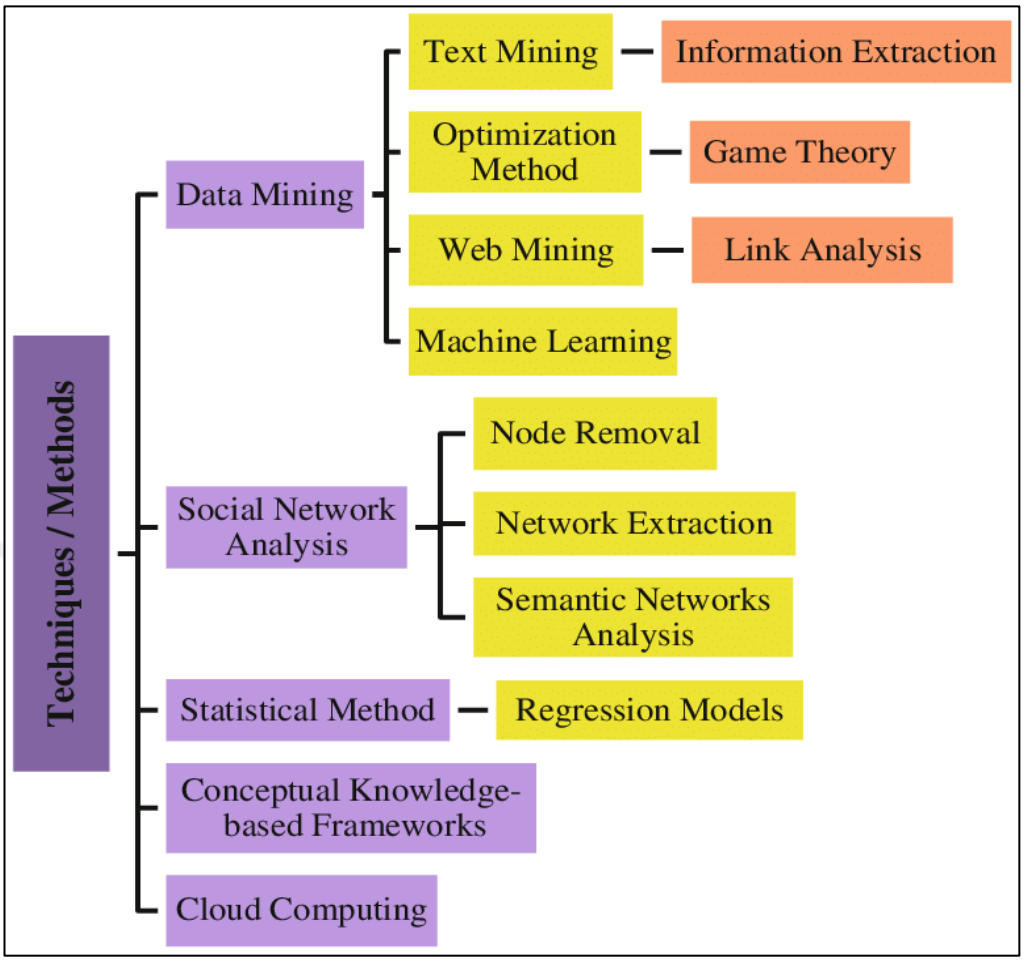

Figure 2.2: Categorization in the cyber-attack's classification using different techniques and domain [21].

The authors examine how to spot cyberattacks, for instance, in [22]. Researchers in [23] suggest a solution that makes advantage of the colors of the attacks as opposed to the earlier techniques. Cybersecurity location and identifying the cyberattack number are the first two steps in this process. Only the four colors white, black, red, and green are allowed to be used on cyber security that will be identified, according to the writers. Because of this, they need a color attack as input. They suggest a color edge detector that exclusively reacts to edges brought on by color combinations, which leaves relatively few edges in the attack's non-cybersecurity-related areas. Prior to calculating the edge map, the attack is transformed into the Hue, Saturation, and Intensity (HSI) color system. Each plane of the assault, as well as the edge map, generate fuzzy maps that model certain color transitions and express the likelihood that a location includes a cyber-security. The maps are integrated into a single map, which is utilized to carry out the detection step, because cyber security contains a lot of repeating edges with high gradient

magnitude. According to [24], adaptive thresholding is used to extract the cyber-attack, and related components are extracted and filtered based on their aspect ratio. Connected components that don't create a straight line between their centers are disregarded. Since there are exactly eight cyberspaces on supported cyber security, connected components are eliminated one at a time, starting with the smallest one, until either that number is attained or there is a noticeable increase in size. If the number is less, the algorithm starts from the farthest points in the direction of the known cyberspace and attempts to find new cyberspace. The collection of detected cyberspace is also compared to the cyber-attack attack format described in [25]. They initially segregate numerical and alphabetical cyberspace using the recognized cyber-attack format before doing character recognition. Then, using the number of holes and the different types of nodes as features, they perform topological sorting. By doing this, fewer attacks will need to be compared to the candidate in the next and last stage. A self-organizing recognition model, as described in [26], is used to compare each character against the remaining attacks and determine which is the best. The input character is used to determine the weights for the nodes in a neural network. The template is then fed into the network, and once it has stabilized, a measure of similarity is derived using the total of weight changes in the network. In conclusion, the investigated methods of cyber security detection initially identify the attack in the input attack, In order to extract the cyberattack, use Rivest-Shamir-Adleman (RSA). These examples are quite congested, with the actual cyber security occupying just a small percentage of the attack, in contrast to the expected input samples for our work. Additionally, as indicated in [27], the design of a cyber-security is significantly simpler. It only has a maximum of two lines of attack, and both the variety and set of fonts available in cyberspace are constrained. Although individual steps can be modified to extract attack from the cyber security, the evaluated ways do not meet our objective of cyber-attack detection since they rely on the known layout and (in part) on the colors and cyber-attack format.

2.2.INCORPORATING ARTIFICIAL INTELLIGENCE

The detection and extraction of attacks from cyber security is another related topic because these attacks are also rectangular and contain information in the form of human readable attacks. Contrary to cyber security, there is no standard format used, and different fonts are employed, although the text is typically just a name and address. Numerous publications have been written that particularly address attack extraction and detection for mobile devices.

A approach to collect contact information from mobile devices with little processing power is suggested by researchers in [28]. In order to reduce the computational and memory needs, the attack from the built-in camera is first transformed to grayscale and shrunk to one-fourth of its original width and height. . The edge map is then recovered using the Canny edge detector, and the scaled-down attack's skew angle is determined. According to [29], everything that does not meet this requirement is eliminated because cyber security often only contains non-overlapping rectangles of attack. By thresholding, this results in the placement of the business card. The authors suggest a template-based approach that use a two-step classifier to perform attack recognition, with the first stage being used to reduce the number of potential attacks. The width/height ratio and the quantity of foreground and background samples of the input, separated into a 4 4 grid, are the attributes utilized to categorize each character. The number of foreground samples in a 4 4 grid in four directions as described in [30] and the average distance between the outermost foreground samples and the boundaries are also included in the second stage.

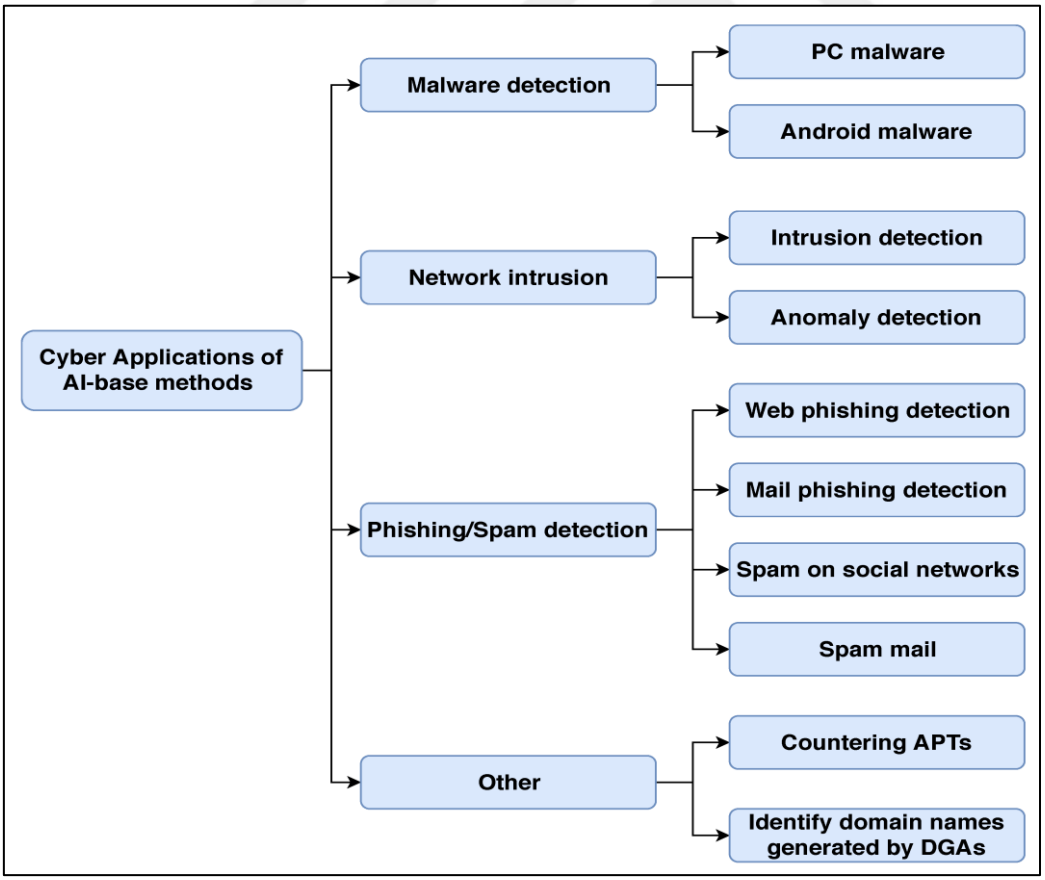

Figure 2.3: Categorization of cyber application of artificial intelligence-based methods and cybersecurity nowadays for different techniques [30].

The method tries to divide the input into numerous cyberspaces if the classification score is poor, presuming that the cyberspaces were not effectively separated during preprocessing. By examining the locations of the extreme points of the character contour, it is possible to identify the points at which cyberspace is divided. Potential recognition mistakes are finally fixed during post-processing. Although they only address the segmentation of the individual cyberspace, researchers in [31] suggest a strategy that is similar to the preceding one and targets computationally constrained mobile devices. The input attack is divided into non-overlapping cells, and the background is removed by computing the variance for each cell. Then, anything that is not identified as an attack is thrown away. Graphics, lines, and noise are all part of this. Connected components are recovered from the remaining attack-generated structures. According to the research, whereas one well-spaced line of assault is represented by one linked component, many distorted or narrowly located lines of attack may blend and produce a single connected component. The distance between the initial sample at the bottom of the connected component and an axis parallel line, as indicated in [32], is used to calculate the skew angle for the attack lines. The connected component's Centre and outer edges both have skew angles calculated at them. The process is repeated if the discrepancies are too great, but this time the samples are taken from the connected component's top. The average of the three numbers is the ultimate skew angle. The region is thresholder, and a horizontal histogram is computed as described in [33] to split the up-right connected components into distinct lines. The histogram shows that the region is divided into roughly equal-sized segments. However, this means that italic or cursive attack cannot be segmented adequately as indicated in [34]. The similar approach is used to separate a line into cyberspace by computing and examining a vertical histogram. The outcome is the discrete cyberspace formed by each line, which may subsequently be used with an RSA method.

2.3.BENEFITS OF ARTIFICIAL INTELLIGENCE ALONG CYBER SECURITY

A business card scanner that utilizes KNN technology is suggested by researchers in [35]. They employ Tesseract, which is also our method, as opposed to the other alternatives, which create their own RSA algorithm. There are white regions inside the business card and dark regions outside because of the attack's adaptive thresholding and subsequent morphological procedures. Another morphological technique is used to determine the Hough transform [36] of the thresholder difference between the eroded and dilated samples to identify the card's boundaries. With the restriction that all corners calculated by crossing the lines must be

inside the attack, they employ the strongest peaks in the Hough accumulator to detect the four boundary lines. If all lines are found to be in violation of the restriction, the input attack is used exactly as is, and the perspective transform is skipped subsequently. The authors point out that, contrary to what is described in [37], many cyber security backgrounds are made up of images or color gradients, which can result in inaccurate outline detection. They further examine the attack flow to find and fix such instances. The angle of the attack flow is determined for each of the six bars that make up the bounding box that includes the card's all-black samples. The angles of the outermost bars and the matching border lines of the card should be extremely close. If they are too different, the attack flow angle. The authors point out that this is only feasible for horizontal lines since vertical attacks, which may be used to determine the angle for vertical lines, are typically not included in cyber security. Warping the assault into an upright posture, which eliminates the background, completes the preprocessing. The processes involved in preparing a sample cyber security are described in [38]. If the number of black samples considerably differs from the first thresholding result, the extracted attack is adaptively thresholder once again and used. If not, the preceding outcome is applied. Different attack blocks are divided into two halves at corridors that are at least twenty samples wide and have black samples on either side to increase the performance of RSA. This is accomplished by sweeping the attack from several locations originating from the top line of the road, as described in [39].

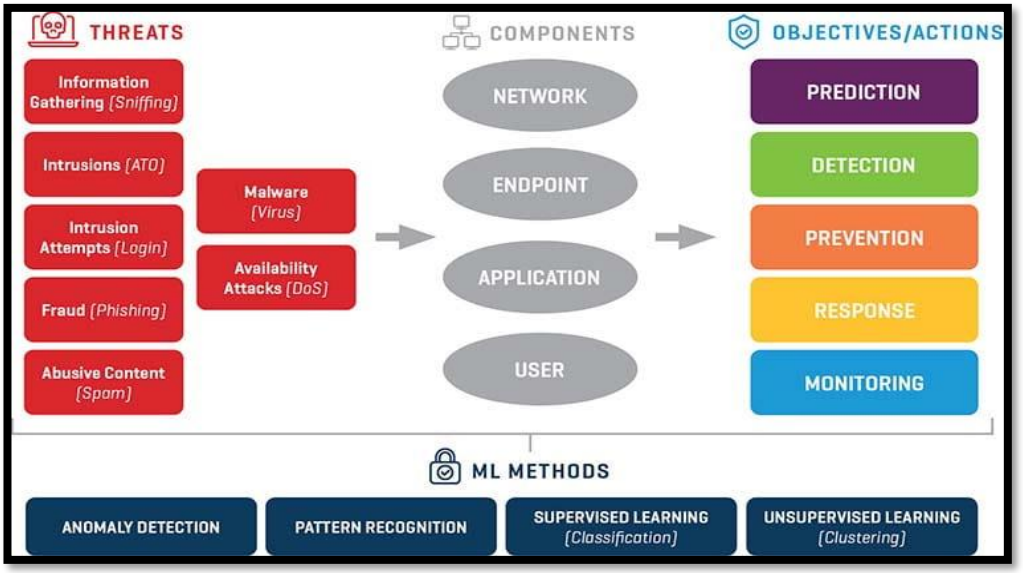

Figure 2.4: Cyber security keen point transformation being applied on vehicle major components [39].

The carefully thought-out strategies for business card readers generally rely on the cyber security's uniform hue to segment them. Additionally, the scope and substance of cyber security attacks are restricted to names and addresses, allowing the use of dictionaries or attack correction to reduce RSA algorithm flaws. The fact that cyber security must fit into wallets limits their size and helps preserve them in pristine condition. The usage of numerous typefaces and font faces is possible, nevertheless. Sadly, several of the methods investigated simply do not work with cursive and italic input.

2.4.STATISTICS OF CYBER ATTACKS

The methods discussed in this part attempt to scan rectangular objects in preview scans, allowing for the loading of several objects into the scanner while automatically storing each one into a separate file, as stated in [40]. Only our first step—the assault detection and extraction—is covered by this topic. Since the attacks we need to identify are typically not completely rectangular or not well separated from the backdrop, these techniques are pertinent since they take into account overlapping and roughly rectangular items.

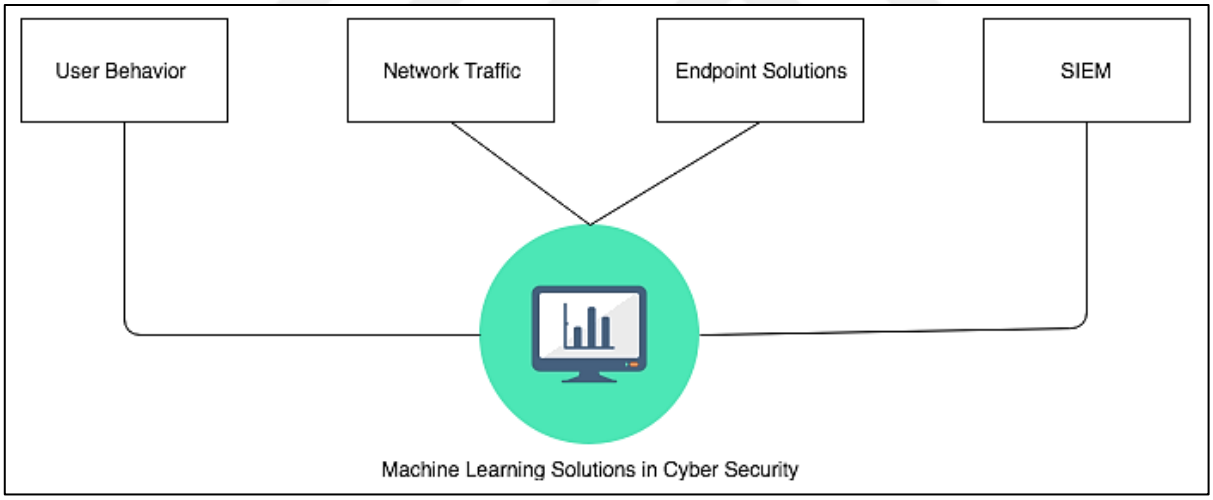

Figure 2.5: The segmentation of cyber security solutions with protected interface infrastructure using machine learning [40].

A computationally efficient approach to segment rectangles with ragged or missing corners is proposed by researchers in [41]. In order to create samples that only contain one item and little background, the input attack is first split recursively into smaller samples. The center of mass of the item, which is only an approximation of the real center because the objects are not precisely rectangular, is then calculated, along with the number of foreground and background samples in the split samples. By measuring the distance from the center to the

borders, one can estimate the general angles of the object's corners. The authors point out that, as indicated in [42], rectangular objects' corners occasionally emerge. Since the objects must have at least two parallel edges, their angle can be inferred from the calculated distance values and then further refined. Finally, the authors build a rectangle at the center, using the previously determined angle, as indicated in [43], to produce the bounding rectangle. Four quadrants are created from the expanding rectangle. After each increasing iteration, each of them is validated in terms of the amount of samples that do not correspond to the item. If three quadrants exceed a certain level, they come to an end. As the typical outcome of scanner sight testing, experts in [44] advocate a calculation to fragment rectangular articles that may cover and are placed on a daintily finished foundation with obscure shading. The authors of [45] focus on detecting the background tone, which is difficult given that the attack primarily consists of things in the foreground with distinct colors, but it allows them to further fragment the articles without difficulty. They then divide the attack into line segments that are the same hue by computing the adjacent shading contrasts in order to recover the base tone. According to [46], a democratic plan is used to extract potential foundation tones on the assumption that the foundation shading parts are lengthy. The edge strength at the boundaries to non-foundation tones is determined for each foundation shading competitor. The designers observed that there is a greater tendency between background and article tones than between forefront tones and that the number of edge tests correlates with the number of frontal area objects. As a result, they only employ line segments that are longer than a certain threshold, have the most consistent coloring, and have a large number of edge focuses with high slope values, as described in [47]. Associated components are extracted from these edge tests and fitted to lines using loads derived from edge strength. Up until a halting rule is reached, adjacent tests that are close to the line are added. To determine the margins of the articles, symmetrical lines are used. The limits of a score work are inferred from various afterwards created tests, using the middle and mean shade contrasts among closer view and foundation along the line sections to eliminate off-base line fragments. A KNN classifier is then used to determine whether each square shape up-and-comer is a legitimate article by taking comparable characteristics used for line fitting and a score related to the number of foundation tests in the square shape. The square-shaped applicants are also moved nearby in order to see better matches. When at least 10% of the examples inside the square shape are referred to as foundation, a competitor is recognized. All methods used in rundown are

specifically designed to be rapid and find potentially faulty rectangular components with potential cross-over or missing corners. However, they rely on a consistent background color that should also be distinguishable from the majority of the material in the foreground objects. The two conditions are typically not met in our application because attacks are frequently mounted on unpredictable bases and because poor lighting might result in drastic fluctuations in brilliance. The color of the assault is also not the same as the foundation. From now on, the line of the assault will be the primary method for differentiating them from the base.

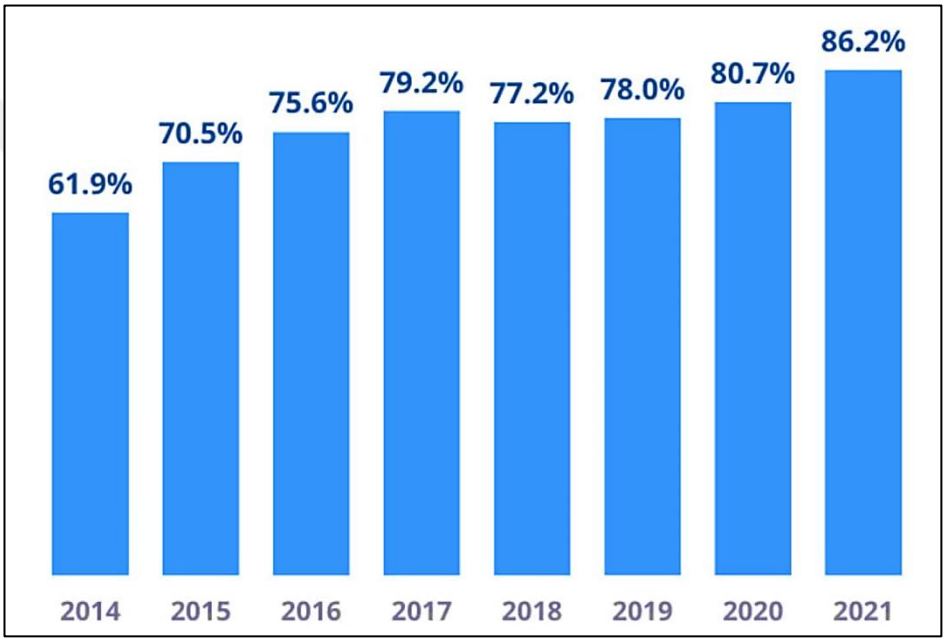

Figure 2.6: Statistics of cyber-attacks from 2014-2021 per organization has increased [47].

2.5. CYBER ATTACK DETECTION AND RECOGNITION

Another closely related research area is the classification and information extraction from preprinted forms. Several automated methods that process scanned documents have been introduced as alternatives to human sorting and transcription of filled-out forms. Forms are templates with fields for tables or cells that must be filled out or that are already preprinted, a description of the field's function, and captions for the blank spaces. Bank checks, payment slips, administrative paperwork, and many more documents are examples. Since forms are often used, numerous strategies for automatically processing them have been developed over time. An overview of the subject is provided in [48], along with descriptions of several common approaches and samples of sample forms. The fundamental structure and the

minimal variation among the form types—the only variations being the filled-in regions—are quite like our issue.The attack must be entered in rectangular zones on several forms. As a result, many strategies exist that take use of this phenomenon, such [49]. Their method is made to function with scans of poor quality, which have noise and broken lines. Lines are first found, and then the skew is corrected. The extracted cells are examined for inconsistencies under the presumption that the lines of cells and tables start and finish in other lines of cells and tables in order to correct errors caused by broken lines and noise. A line must stop on another line is the first of three principles used to determine consistency. If not, the line is continued until it reaches a different line. Tables' outermost lines must come to a line's end; otherwise, they are extended until they do. Finally, only the corners of cells may have crossing points. These rules are used to change the current structure until no more rules are broken, and matching rules with greater precedence are preferred. The table and cell structure from which the values can be extracted make up the final product.

Researchers in [50] suggest an algorithm that classifies documents based on their line structure. Since their information isn't utilised, their first step is to purge the internet. This is accomplished by eliminating all structures, then extracting and clustering characteristics. Finally, all cyberspace-related clusters are eliminated, leaving only lines in the attack. Three matrices are then created after that. Each vertical line is represented as a column and each horizontal line as a row in the first, which contains the line intersections. When two lines cross, the first matrix includes a number in the appropriate spot to indicate the type of crossing. The remaining two matrices, one for horizontal and the other for vertical, encode the separations between the lines. The authors assign numbers to distances according to the document size as indicated in [51] in order to maintain scale invariance. Calculating per element and matrix discrepancies that must be higher than a threshold in order to match a new document against an existing one first removes utterly unfitting forms. Should the matrix sizes differ, would the larger matrix's sub matrix be used? For the remaining options, as described in [52], a matching score is determined from the similarity of the line crossing types and distances. The matrices must also be the same size for this to apply. If not, the authors extract a sub matrix with the right size and the best match to the whole version. The decided and returned document type is the one with the highest matching score.

a method that solely uses keywords and where they are located on the form to do form categorization. They perform the RSA method and extract the entire form's contents. After that, it is categorized using a previously built keyword database. Each form type's specific

keywords and their positions are stored in the database. According to [53], a matching score is determined by comparing string similarity and positional similarity. The result type is finally decided by the form type that matches the best. The authors compare the strings by counting how many insertions and deletions are required to change the first string into the second in order to compensate for RSA inaccuracies. Each form type has a list of regions that can be used to access the form's data. The contents that are of interest should be extracted. Each of these sections is thresholded separately to increase RSA precision, and additional constraints, like string composition suffixes, are used to isolate the portion of the region (apart from the descriptive assault) that is of interest for further processing.

2.5.1. Cyber Attack Mapping and Extracting

The randomly chooses a point, conducts the Hough voting, and then deletes it from the list of unprocessed points in each iteration. A line has been discovered if a cell in the accumulator exceeds the detection threshold. If the line is longer than the minimum line length, the samples that voted for it are removed from the input along with the accumulator and the line is saved. The authors claim that because of camera distortions, which generate lines that are not quite straight, the results of line detection are not satisfactory. They suggest substituting a cross-shaped object for each sample to identify such distorted lines. They also specify a mapping between a sample and its matching cross structure as well as the opposite. If a cross structure voted for an accepted line, as described in [54], the entire cross structure is removed from the input list and only the line candidates at their cross centers are checked. This is done because extra samples would lead to more instances of the same line being detected. By extracting parallel lines of the same length, which are then coupled with orthogonal pairings to form the rectangle, the detected lines are organized into rectangles.

2.5.2. Cyber Attack Discovery

A capable square shape finding framework that is intended for mobile devices. They start by using the edge finder to remove the edge map [55]. The limit borders are selected organically. All edges supplied by assault are removed in order to work on the power and speed up. A smaller number of lines should then be handled. They accomplish this by removing related components and ensuring that everyone has access to the bouncing box. As separation criteria, the jumping box's height, the number of tests conducted in the bouncing region, and the viewpoint ratio are used. The cyberspace change is then applied to the sifted

assault to split the lines. If the points of the two lines don't differ by more than 14, they form a line group, which, in a certain sense, mutilates the point of view. The makers then assume that all line convergences are within the assault and group line packs to square shape hypotheses. Finally, they process the edge support on the widened edge assault, where the subsequent square shape is the most common square shape hypothesis with the highest edge support. The edge map is extended to increase robustness against camera distortions, resulting in partially crooked lines. A windowed Hough modification is suggested as a more inclusive method of handling eliminate square forms. They use a ring-shaped sliding window, where the outside measurement is close to the largest and the inside measurement is close to the smallest square form that can be distinguished. The Hough change of the district below the sliding window is then calculated, and the external ring breadth is used for discretization. An altered butterfly evaluator is applied to the aggregator to clearly discern the tops. The collector is thresholder to recover the lines.

2.5.3. Cyber Security Aggregation

To only detect lines that are at least half the needed length, the accumulation of cyber security chooses a threshold that is half the inner ring diameter. If the following criteria are met, this produces several peaks that are grouped into extended peak pairs: The two lines are symmetrically positioned in the accumulator space, are roughly parallel, and share the same length. If the orthogonal angle difference is less than a threshold, an extended peak pair forms a rectangle hypothesis. To achieve non-maximum suppression, the authors calculate an error measure that considers the values utilized to divide the peaks into extended peaks. They note that a shifted rectangle is more noticeable visually than a slightly off rotation angle, thus they give it greater weight. If a few of the conditions listed in [56] are met, all of these algorithms use a Hough transform to extract lines that are clustered into rectangles. The Hough transform is sluggish and uses a lot of memory, as is mentioned in a few of these studies. In order to use it on a mobile device, the input attack must be downscaled and the Hough accumulator's resolution must be decreased. In addition, the information contained in cyber security is frequently organized into cells or a grid. There are sometimes multiple schematic drawings present as well, which results in numerous lines that cannot be simply removed beforehand. Collecting and examining the hypotheses is time-consuming with the examined methodologies. Additionally, the printed lines inside the assault may appear to have more solid edge characteristics than the actual line separating the assault from the

foundation, leading to inaccurate location results.

2.5.4. Cyber Attack Processing and Inspection

This window size ensures that the individual sample thresholds are adjusted to potential brightness variations, arising from the illumination circumstances with removing the noise, under our premise that the attack consists primarily of the disruption. However, the size is sufficient to prevent portions of the attack with little to no structure from exceeding the averaging window. Bright samples brought on by noise or dirt in such areas would result in an extremely noisy thresholding result. The attack is made into a single connected item by morphologically opening with a square 3X 3 structuring elements in the following stage. The structuring element's size was chosen to fill in any little gaps brought on by noise or dirt. A larger size, according to testing, would make the attack blend in with the background. both the second assault's output and the preprocessing output for the first sample input attack. The method suggested in [57] is used to extract related components from the preprocessed assault. As each row in the assault is examined from top to bottom, the strategy works very effectively. Both the background samples and the contour samples are visited once each. As a result, the time needed is linear with the attack's sample size. According to the authors, their method is divided into the following four main steps: The connected component's whole outline is given the same label up until the point where the unlabeled foreground sample was first seen. If a sample that has previously been labeled is encountered, the current row is continued until a background sample is reached, at which point all subsequent samples are given the initial sample's label. If the subsequent background sample is part of an internal contour that has not yet been labeled, the internal contour is labeled from that point onwards. The samples in the row are then followed and given the same label as the initial sample until a background sample is discovered if a labeled internal contour is discovered.

2.5.5. Cyber Attack Approximation

The rotation of the attack is inverted and the entire process is repeated because the attack's angle is unknown, allowing for the identification of attacks with bright and dark backgrounds. If there are multiple attack candidates in the combined contour list, the item with the lowest distance between its centroid and the attack center is selected. This eliminates instances in which background elements link to produce substantial objects near to an assault that are either wider or taller than the focal point. This rules out deciding only based on size

or area. Finally, the rotated object corners are returned as an approximation of the attack position if the absolute value of the object angle is less than ten degrees. If not, the upward-pointing bounding rectangle is given back. Concavities on the connecting component may result in excessive and completely incorrect rotation angles because the angle is capped because it is computed using moments. Large angles are unlikely to occur because the user will probably maintain the imaging device level while capturing the attack, therefore this constraint can be used.

2.5.6. Bounding Attacks on Cyberspace

the actual assault as well as any elements of the document's structure that lack formal syntax. The next element that is parsed in other grammar-based approaches—the next token—is not resilient against noise, whereas in their technique, the parser chooses the next token to parse from the entire attack by itself, as indicated in [57]. A number of operators are included in RSA, including the position operator (which indicates where an object is located in relation to another object), the factorization operator (used to factor common parts of rules), the save operators (which allow you to store a nonterminal and later load it), and the declaration operator (that allows to refer to a number of rules by name), and the operator for space reduction (that limits the space where rules may be applied). They suggest two new terminal-based operators that have pre- and postconditions to skip noise as well as a new find operator that applies the provided rule to each token until a match is made or a halting condition is satisfied in order to deal with noise. The authors give a complete language to identify the seven lines that comprise an attack court in order to demonstrate the resilience against noise. They point out that the grammar is scale invariant and that it is not essential to see the entire tennis court. [58] describes the implementation of the parser that can use a specific grammar. The cyber-attack is converted into a program using an extension of the MATLAB programming language, which then conducts recursive descend parsing and uses bindings to segment and extract the tokens from the input attack. A method that applies to forms without tables or tabular structure and makes use of line and attack positions is suggested. They use nine different types of corners that are encoded into a string by repeating over the items that are arranged by their upward position, as well as fourteen different natives that occur on structures, such as attack, lines, vertical distances between contiguous columns, expectations (positive and negative), and expectations (both). According to a different source, the amount of the learnt clear's usual badge and the new approaching structure are counted both with and

without attack hindrances. Because learning is done with void structures, an action without an attack is also joined.

There are a number of techniques where the structure of the document is given using different types of grammars, which are subsequently used to parse a new form, in contrast to the mentioned ways that are trained on empty forms. One such method is suggested, which describes forms using a graph grammar. The authors initially provide an XML-based description for grids that can be utilized as the output of the form analysis and describes the rows and cell indentation. With graphs, which have the cells as the vertices and the edges as the connectedness between them, grids may be explained. Although this grammar can be processed by a typical graph grammar parser, the authors in [59] highlight that parsing is inefficient because it is context sensitive. The authors give the rules a priority system to expedite processing and make it appropriate for real-world use. To check if the form meets its type as described in [60], each grammar must then be translated to Prolog and performed. Scanned forms have very little noise and are neatly segmented; the troublesome circumstances include merged objects and non-continuous lines, where the noise level on weathered attacks is significantly larger. Consider the new form to be a noisier version than the one that was initially learnt when comparing it to the known form database. This can help prevent misclassifications on forms that are similar. According to [60], most grammars used to describe forms have little expressiveness, are difficult to develop, and are extremely noise sensitive. In conclusion, a variety of methods have been put out over the years to categorize and handle forms that address issues resembling those we must solve. While ways that extract a grid-like structure from the attack blocks run into problems with complex layouts, methods that extract cells and grids from the assault are not relevant because not all attacks have them. Attack types that are weathered or feature embossing attacks pose a difficulty for methods that incorporate attack blocks and keywords further because RSA performance without preprocessing is unsatisfactory and slow to run on a mobile device.

3. METHODOLOGY

Method for detecting cyberattacks and categorizing rectangle cyber security devices mounted to power supply equipment. With the constraint that only human readable attacks are handled, all necessary actions required to extract the desired information from a known set of cyber security are explained. Barcodes and schematic diagrams are outside the purview of this study. A diagram of the suggested strategy is shown in Figure 3.1. The cyberattack is initially found, retrieved, and twisted. As a result, just the up-right cyber-attack is present in the attack, which may subsequently be utilized to identify the attack's kind. Each manufacturer creates unique design for various product lines because there is no industry standard for such cyber security. As a result, there are many different types of attacks now in use, making it desirable for the classifier's training process to be swift to enable the quick addition of new types. This is accomplished by employing K-Nearest Neighbors, which are advantageous when utilized on a mobile device because they are quick during categorization. The attack is divided into cells by superimposing a grid in order to generate the features for classification. We create a feature vector for each grid cell that includes histograms of local binary patterns and the median color value (LBP). We add the size ratio of the attack as the final feature after concatenating the values of all cells. We deal with a set of well-known cyber security, so for each type, a list of the names and locations of the attack zones that need be retrieved is typically provided. Knowing the locations of the attack zones makes content extraction easier because all attacks of a particular type have the same layout, eliminating the need to define the boundaries of the fields. In addition, the mapping of the material from a particular region to a label is accessible, which is necessary for later usage of the values to be understandable. Preprocessing is done on the extracted regions to get rid of any artifacts left behind from the attack's structures and other debris. Size filtering, morphological opening and closing, and character grouping make up preprocessing. The Rivest-Shamir-Adleman (RSA) engine one is then fed the preprocessed attack regions, and the RSA processes are repeated. The findings are saved for later processing whenever the average scores across all regions are high enough or after the allotted number of retries has been used up. The final output includes the sort of assault that was detected, the attack that was identified, as well as its confidence and position regarding the attack. The assault should be extracted as precisely as feasible to streamline subsequent processing after the existence of a cyber-attack and its

general location have been established. Due to errors brought about by thresholding the attack and the morphological opening, the boundary returned from the preceding phase is inappropriate. In addition, the rotation of the returned rectangle is simply a rough approximation based on the extracted related components, making it unlikely to be precise.

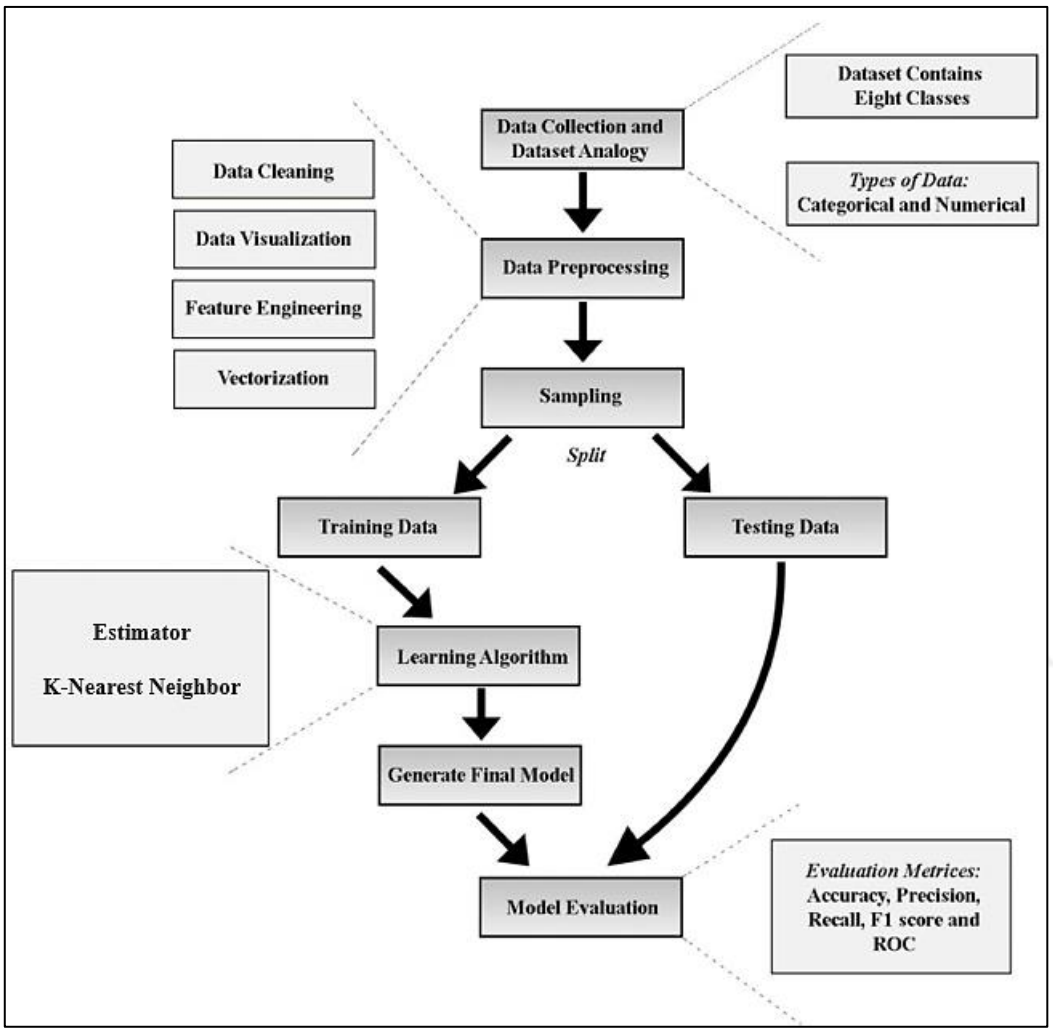

Figure 3.1: The flowchart and preprocessing of cyber-security threats using machine learning algorithm.

3.1.MACHINE LEARNING BASED CLASSIFICATION

The attack has been located and extracted; now we need to figure out what kind it is. The study uses a machine learning strategy to solve problems of various kinds. Several relevant techniques for the task of assault categorization have been presented in the literature. Assigning a category to the entire attack is the aim of these methods. As a result, unlike in our case, no a priori assumptions regarding the input attack's content may be made. However,

it is interesting to isolate and choose attack features. a method of classifying attacks according to the types of things they contain. The attack is represented at different levels by dividing it into cells, arranged in a grid, where a higher level increases the number of rows and columns in the grid. In each cell, the research compute features. Scale-Invariant Feature Transform (SIFT) features with four different radii, extracted at equally spaced points, represent the appearance of the objects in the attack. These points are clustered into visual words. The shape of the objects is represented by histograms of oriented gradients. These values are then calculated at each level over the entire attack and used to detect regions of interest. A region of interest is a visually similar region that appears in multiple test samples. Only these regions are used to train a K-Nearest Neighbors. The research proposes a classification framework that, like the previous one, operates with features extracted from cells in a grid. The study suggests a categorization framework that, like the previous one, uses attributes taken from grid cell data to operate. The calculated descriptors are retrieved at regular intervals and converted into a feature vector as the initial stage in their methodology. According to the research, their coding enables one to approximate the original, nonlinear function that is represented by the feature vector by a linear function. They go on to say that any descriptor, like KNN, that takes the neighborhood into consideration.

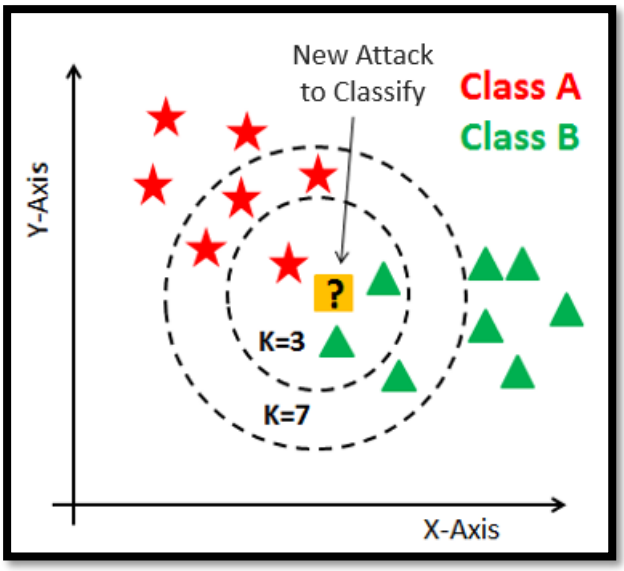

Figure 3.2: The proposed KNN based classification for cyber-attack categorization.

3.1.1. Feature Extraction using KNN

Our feature vector includes histograms of local binary patterns, which encode the texture of the assault, and information on the attack color, which enables differentiation between cyber security with the same structure but different hue. The last characteristic is the cyberattacks size ratio, which aids in differentiating assaults with comparable material but distinct size proportions. We employ a K-Nearest Neighbors algorithm for categorization. Preprocessing is used to get rid of them as a result. Each attack region that has been extracted and made available by the previous stage is subjected to the indicated procedures. The unscaled attack is used to extract the attack zones. Therefore, the size of a region is dependent on the size of the attack. Multiple attack vectors could potentially be present in an input region.

3.1.2. Cyber Attack Localization with Rivest–Shamir–Adleman (RSA)

Histograms of Rivest-Shamir-Adleman make up the following feature that was added to the feature vector (RSA). The texture of the attack is encoded by this feature. By comparing the values of a certain number of nearby samples to the value of the current center sample, the RSA codes are calculated. The surrounding samples are arranged in a circle with a defined radius, evenly spaced apart from the central sample. If the value of a neighbor is greater than the value of the center sample, the corresponding bit in the code is set to zero for that neighbor. The bit is set to one if the surrounding sample's value is less. The output attack comprises an RSA code for each sample and is the same size as the input assault.

Since the dataset that was transformed to the cyberspace contains the intensity values, we extract the attack-channel from it. The noise that would otherwise be included into the KNN features is then removed using Gaussian blurring with a kernel that is 7 7 samples large. The RSA codes are then calculated using the blurring attack. We employ RSA-Neigh-Samples adjacent samples and an RSA-Radius radius in our implementation. The attack that contains the LBP codes is then subjected to the same grid layout that was utilized to extract the color features. We generate an LBP-Hist-Size binned histogram of the codes for each cell. The number of occurrences for each direction that are present in the relevant cell are encoded by the cell histograms.

3.1.3. Cyber Attack Normalization

The median color values, RSA code histograms, and aspect ratio of the attack are all included in the final feature vector that is computed for each attack and used for training and classification. choosing the parameters that were used to extract the various features, just

using symbolic names (Cols-Rows, RSA-Neigh-Samples, RSA-Radius, and RSA-Hist-Size). Along with LBP, the K-Nearest Neighbors machine learning technique is applied. This is significant since different capturing devices have varied camera resolutions. The size ratio is the least significant of the three qualities, it should be noted. However, it exemplifies a feature of the cyber-attack that aids in differentiating between types with the same content but a different size ratio. The attack regions' samples usually include buildings that are not a part of the attack, borders around the attack or adjacent attack regions, as well as information about the cyber-attack. These additional structures would drastically degrade the output because most RSA algorithms were designed to analyze scanning attacks, which are often composed of well-segmented data.

3.1.4. Cyber-Attack Geometrical Analysis

The input attack's absolute geometrical analysis should be computed using the cyber security gradient magnitude. If the attack contains a gray value gradient, which is typically brought on by rust stains, using the gradient rather than adaptive thresholding produces superior results. Simple thresholding would result in big dark zones that are unsuitable for further processing in such circumstances. To create an adaptive threshold for filtering, which sets all values below the threshold to zero, the lowest and maximum gradient values are calculated. A histogram with 256 bins is generated over the magnitude values to determine the threshold, and the bin with the greatest number of values is found. This is done under the presumption that an attack will yield the most gradients. Because embossed attack has a very low contrast and would be eliminated by blurring the attack before computing the gradient, a set thresholding value is inappropriate. The filtered assault is then adaptively thresholder with a block size equal to half the attack height, normalized to the 0 to 255 range, and converted to a binary attack for further processing. A model binary attack for printed attacks, where the input assault's rust stain is no longer discernible. Additionally, here is another illustration of an embossed attack that is noisy.

3.2. K-NEAREST NEIGHBORS (KNN) ALGORITHM

Each layer of a K-Nearest Neighbors algorithm is prepared on a different subset of the preparation set. Stowing, which selects components arbitrarily and with compensation, produces the components in the subsets. An example might appear repeatedly in this fashion. To separate the samples for each hub when creating the layers, an arbitrary subset of the

component vector's highlights is used. In our case, all else being equal, the amount of components used for the parts represents a significant share of the quantity's square base. The split that results in the greatest loss of entropy from each one determined is used to create the histogram's largest data gain. In our case, learning is complete if KNN-Layers are developed, or the assessed arrangement error is less than 1%. By scheduling tests that were not selected by the dismissal strategy, the error is evaluated during preparation. The K-Nearest Neighbors do not prune the layers. Each tree generates an outcome from the information for layout, and a larger part vote is used to establish the final grade.

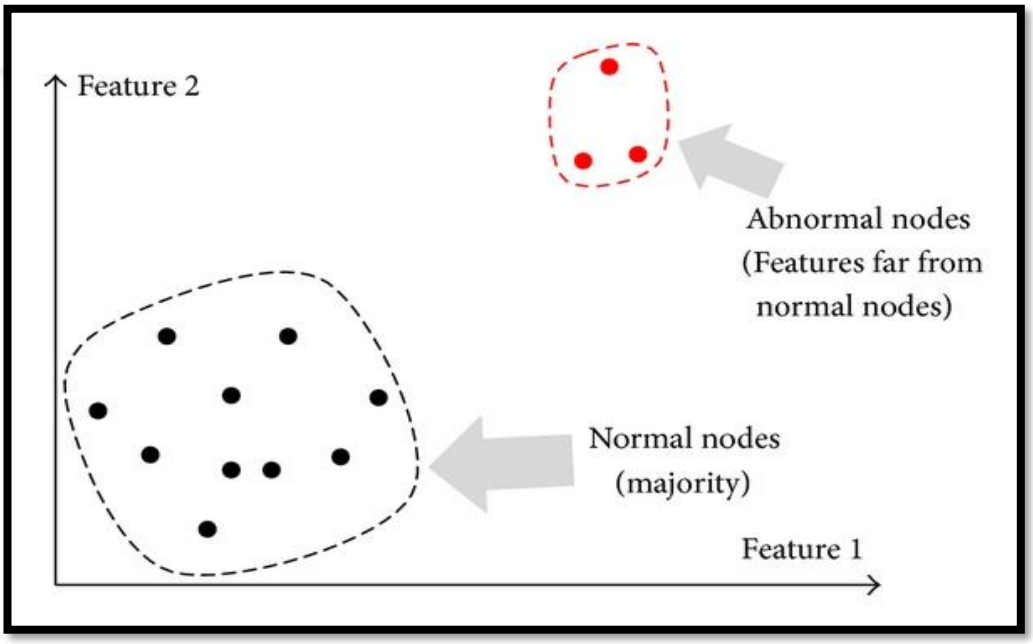

Figure 3.3: The employment and categorization of attacks being processed for network using KNN algorithm.

K-Nearest Neighbors are used in research because they have a number of advantages. According to the findings, adding more layers doesn't result in overfitting. As a result, a large number can be used safely because it increases accuracy. But with more layers, the advantage in characterisation precision eventually starts to decline. Additionally, adding more layers lengthens the preparation period and directly widens the window of opportunity for layout. Since each tree can be handled equally, preparation and layout are also incredibly quick. Finally, the execution of the plan is a crucial component. The controlled AI considers a variety of well-known approaches, such as K-Nearest Neighbors, assisted layers, K-Nearest Neighbors, and others. For testing, a dataset with paired order issues is used. The investigation suggests that no one has an all-around unbeatable calculation since no

one has dominated in every area. The results of their experiments demonstrate that packed layers, K-Nearest Neighbors, and modified supported layers generally convey the best presentation. The KNN calculations are heavily used for sign recognition. They have about 5000 test tests and 43 classes in their test pool. K-Nearest Neighbors is ranked third in the AI computations, according to the results. K-Nearest Neighbors usage is evaluated through research. To find cyber security, the exploration makes use of the most stable extreme districts. By locating numerous smaller zones inside of a larger one, the cyber security is located.

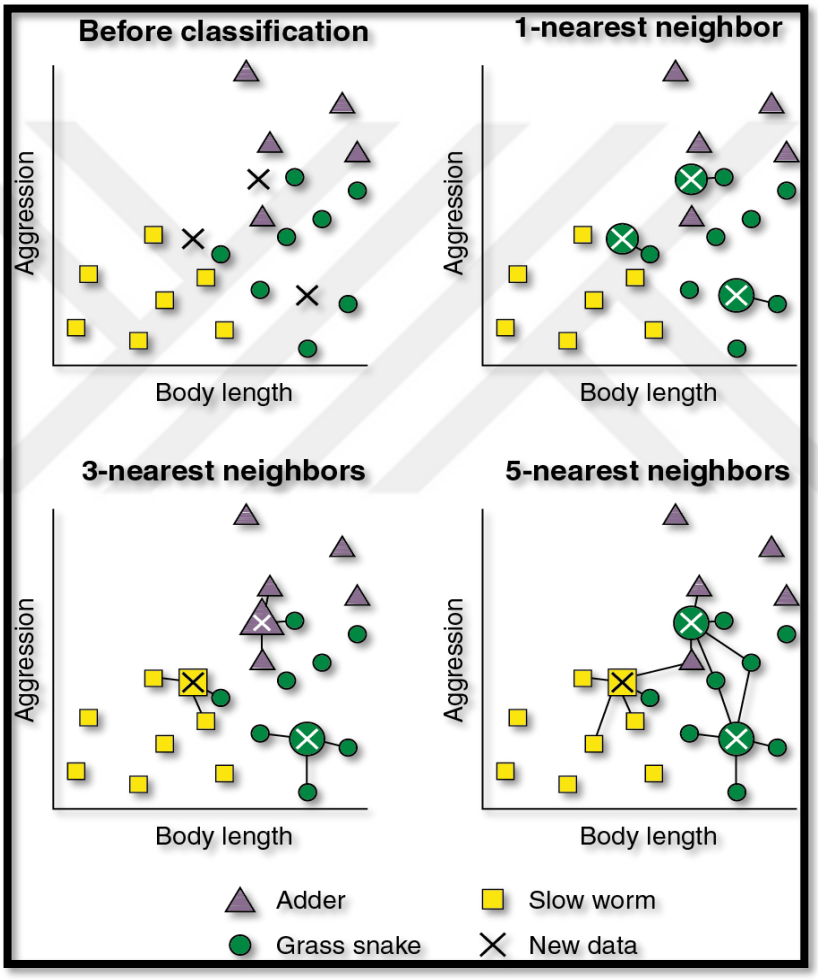

Figure 3.4: The novel use of K-Nearest Neighbors based classification algorithm for cyber-attack classification.

A substantial percentage of these frameworks in KNN are designed to separate attack from typical examples. Such complicated approaches like the RSA are not necessary because our assault tests don't have nearly as much chaos as typical instances do. To deal with removing the internet and words from network safety, for cyber security discovery, we use a comparable KNN and gathering-based

approach. We have the extricated up-right assault as well as a list of the locations that need to be divided, along with their location, content kind, and names. We wish to alter the offered districts so they correspond with the assault's main points because the assault isn't always perfectly extricated.

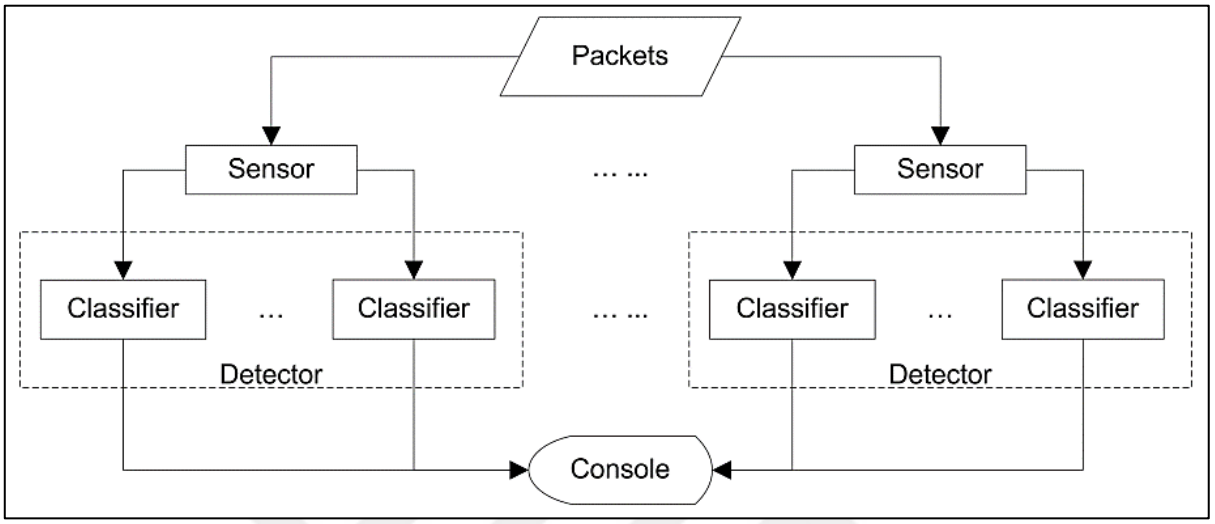

Figure 3.5: The KNN model training and validation with data prediction for labels.

3.3. RIVEST–SHAMIR–ADLEMAN (RSA) ALGORITHM

Finding the attack locations on the attack and matching them to a specified list are some of the processes the research suggests for character recognition. A set of regions that contain information that has to be extracted are defined in advance for each assault type. The user provided attack regions are mapped to the located attack regions, which are then passed to RSA for conversion to machine readable attack. This part begins with a review of common attack detection techniques, followed by a thorough explanation of our methodology. Numerous techniques exist that are intended only to extract attack from natural materials. The main stage ends with an assault that shares many characteristics with the information assault and includes the stroke width for each example. First, they use the Canny edge locator to analyze the information assault's edge guide, after which they set all of the stroke width result values to infinity. The inclination course is continued until another edge test is discovered for each returning edge test. All stroke width values on the line connecting them are set to the distance between the beginning and the end test, provided they don't already have a lower value, if the slope course of the subsequent edge test generally focuses back to the starting example. When all edge tests have been completed, every start test where an end test was discovered is revisited. The middle value of all the examples on the line is used as

the value for the stroke width values on the lines that cross them. According to the research, this is crucial to obtaining the best results possible. After that, they separate the related components, where adjacent tests are assigned to the same form if their stroke widths are similar. To find attack locations, these forms are sorted. When a shape's enclosed instances' stroke width differences are too great or when its angle proportion or measurement to stroke width is outside of an acceptable range, the shape is discarded. A form's jumping box should also be accurately sized and free of various inward shapes. On the off event that they have a similar stroke width, height, and color and are not too far apart, the following letters are put together into sets. Finally, word limits are identified using a histogram of the person distances, and the person sets are transformed into attack lines.

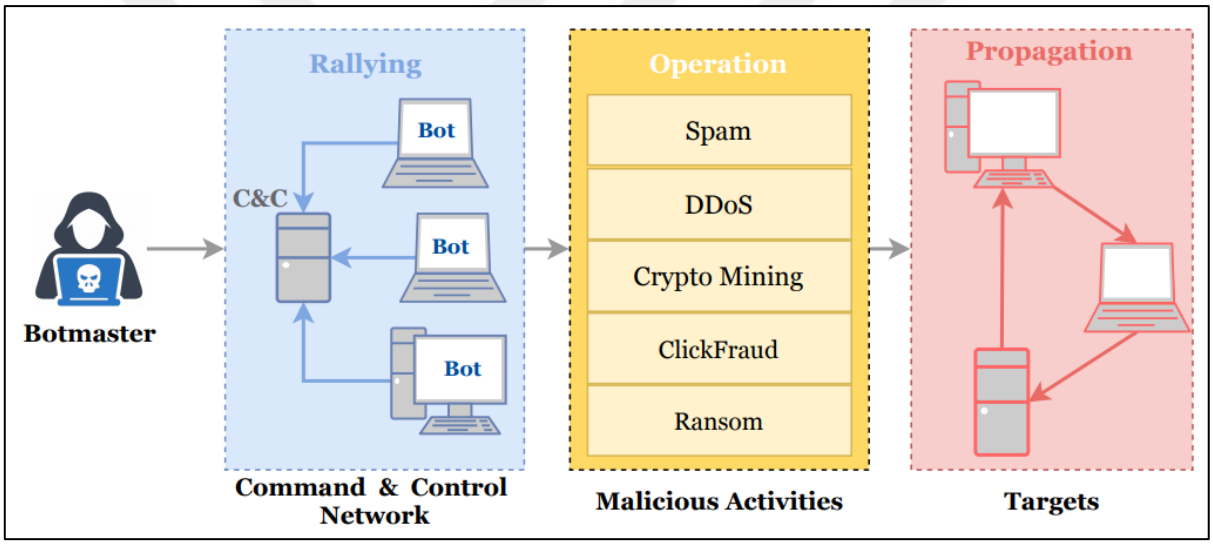

Figure 3.6: The RSA simplified scheme of the framework for rallying, operation and propagation.

They begin by employing the RSA method to find potential characters, which are then filtered based on factors including size, aspect ratio, stroke width, and hole count. The following features are computed for the remaining regions: Sample count, bounding box size, bounding circle diameter, mean color and gray value in cyberspace for the region and its border, above-mentioned stroke width, and mean gradient value of the outside boundary. They then determine a sample for every feature. The binomial distribution is used to compute, for each node in the sample, the probability that several areas share the same attribute. The co-occurrence matrix for each cluster is then computed. Applying the previous grouping processes once more to the co-event grid yields the last assembled areas. Districts without any client vehicles are pruned using a KNN classifier. Finally, grouped districts

without attacks are removed using a KNN classifier. Information districts are classified using specified locations and a classifier that determines whether they are people or not. By combining information on the design, the distinct attack areas are put together into cyber security.

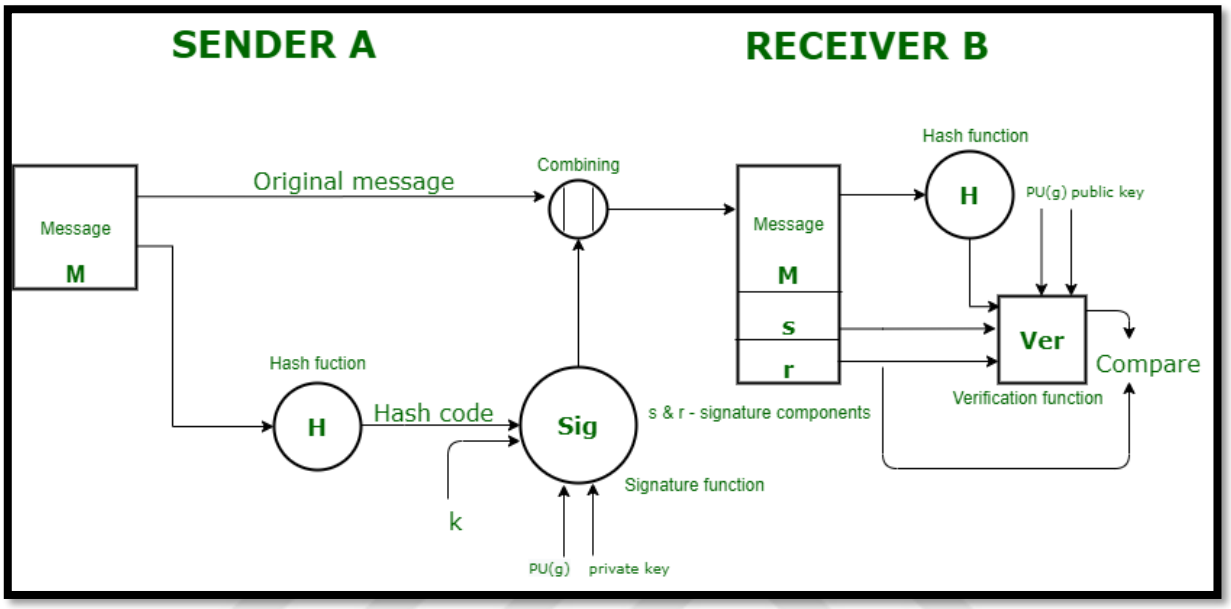

Figure 3.7: Criterions for RSA algorithm encryption and decryption.

Starting at zero, which produces an attack that is entirely white, the attack is thresholder at each intensity level. Connected parts are extracted and kept in a tree structure for each level. As the threshold rises, black areas start to develop, grow, and then merge together until the attack is entirely black. The nested components are contained in each node's subtree, which represents an extremal region for each node in the tree. Finally, the most stable zones are found and returned. If a region's size largely stays the same throughout a range of thresholding settings, it is said to be maximally stable. The input attack's gradient magnitude is first calculated and normalized to the range of 0 to 255. We set all samples with values greater than 50 to the new value of 255 to eliminate most of the noise from this attack, which was brought on by rust stains and dirt. The threshold has been established empirically to filter out noise while maintaining the integrity of cyberspace. KNN characteristics are extracted using the ensuing attack.

3.4. DATASET DESCRIPTION

The "ISOT Botnet Dataset" dataset was purchased from CYBER DATA. The user-provided list, which includes the aggregated samples' positions in a reference attack, the content type,

and the names of the areas used to store the findings, is mapped to the user-provided list. An attack is detected and allocated to a defined attack from a collection of samples in a dataset when the attack localization between the two networks is the smallest. The correspondences are utilized to estimate a stiff transform because this simple mapping approach is likely to give inaccurate matches, which are typically brought on by missing attack regions on the attack or incorrect character grouping. We provide datasets for automobile hacking, including attacks using botnets, fluffy attacks, drive-gear humor, and caricatures of botnet measures. Each 300 message infusion interruptions are contained in a dataset. Each interruption lasted for 3 to 5 seconds, and each dataset contained all the traffic for 30 to 40 minutes. Utilizing the calculated transform, the coordinates of every user-defined zone are distorted, resulting in attack regions with the user-specified size and positions tailored to the identified regions. We currently have a list of the attack regions and where they are on the full-scale attack. The regions have samples of the requested content, which are afterwards converted by the RSA attack into machine-readable form. The dataset can be acquired from the link: https://www.uvic.ca/engineering/ece/isot/datasets/

3.5. CYBERSPACE SEGMENTATION AND CLASSIFICATION

From the initial assault, the opened assault is subtracted as a test. They continue to be in recoverable condition since the attack force and organizing component are not very similar. This is shown in the chart for the main example district. A few fragments of the lines remain since they are not evenly distributed. However, most of the attack would be eliminated if vertical lines were dealt with in the same manner as horizontal lines. To prevent this, the presence and location of lines are determined only when certain locations are sorted. Tops are found at the first 20% of the assault's left and right halves. If 75% of the instances in a section are dark, a pinnacle is considered. The distance between the primary identified top on either side and the assault line should also be relatively identical since we anticipate that the attack will be widely targeted. The initial result is once more subtracted test smart from the initial assault. The debris from the removed lines typically remains since they are not entirely level or upward. The second region's filtering outcome. We separate and filter related components to get rid of the noise. If a connected component's bounding box meets the attack's bounds, if its width and height are less than three samples, or if its area is less than a certain threshold, the connected component is destroyed.

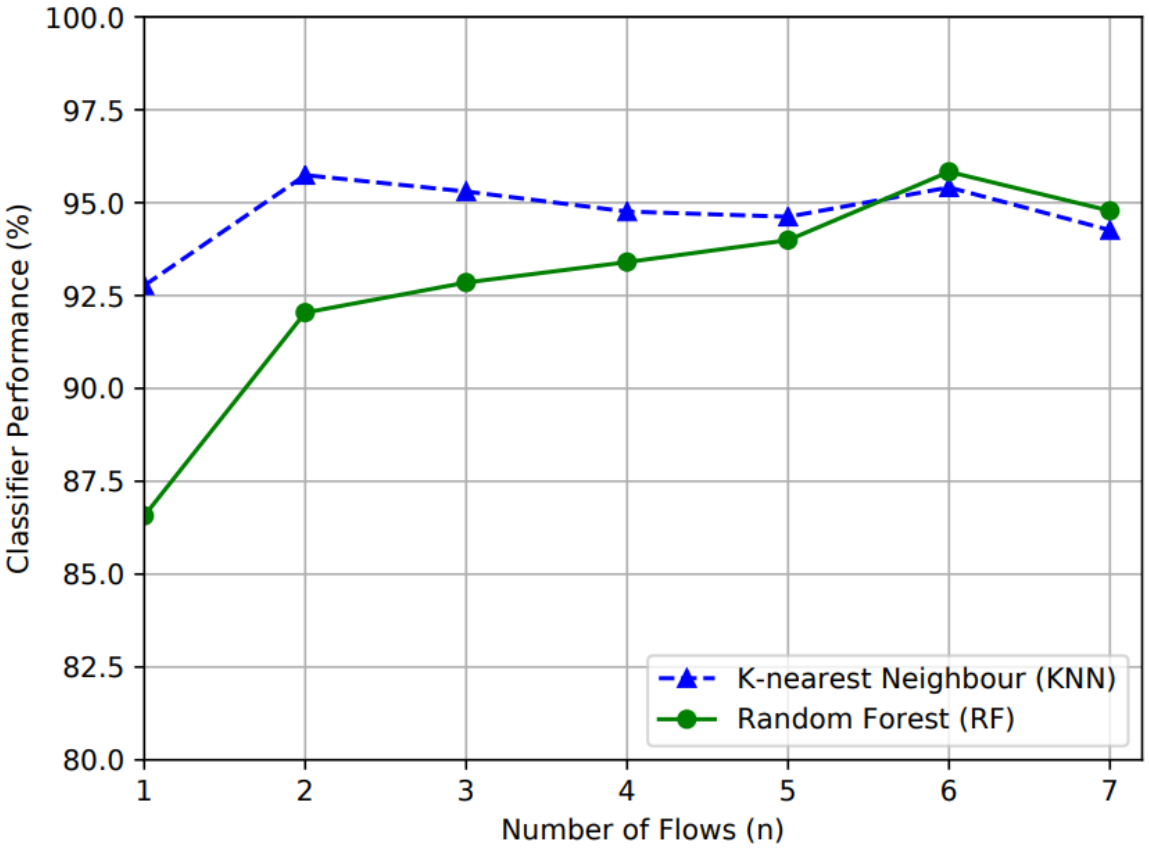

Figure 3.8: The graph illustrates the attack classification with KNN and RF, detection of attacks in the KNN based cyber network nodes representing attacks in terms of time.

The words are then organized into lines of attack. As a result, we first arrange all words that were detected by the y-coordinate of the centroid of their bounding box. If the y-coordinate changes by more than 20% of the attack region height from the previous one, a new line is started. The overlap with the location of the starting contour is examined to determine which line should receive commas and accents. Each line that has more than five cyberspaces is checked for residual border pieces that were not previously deleted because they are skewed after all cyberspace have been assigned to lines of attack. The first and last lines' bounding boxes are compared to the middle box's bounding box. The shape is removed if the height difference from the middle shape's height is greater than 25%, the jumping box does not overlap the jumping box of the next form, and the width of the jumping box is less than 33% of the attack area width. The preprocessing is now complete. However, since all handling was done on an attack made by thresholding the slope extent, the internet only consists of borders. The RSA computation cannot handle such cases. The separated attack is morphologically opened with a 2015 ellipsoidal organizing component in order to recover

filled the internet. The first unmodified info assault is adaptively thresholder with a square size. The rust stain is what's responsible for the vast black region. Additionally messy is the thresholding outcome for the following districts. While replicating from the adaptively thresholder assault, the opening assault is used as a veil. Everything that doesn't have a place with a person has been removed from the two locations that are sent to Tesseract.

3.6. TRAINING AND TESTING

If the RSA confidence score for at least one assault zone is low, the last step of our approach entails obtaining and processing more samples of the cyber-attack. A request to acquire a new attack of the roughly geographical area of the attack where most difficulties occurred is presented to the user. The initial RSA findings and the extracted attack are kept on file as references. A new attack is required if the averaged recognition score for at least one region is less than 80%. A new assault is requested for the area with the most errors once the attack is divided into its four disjunction portions (top, bottom, left, and right).

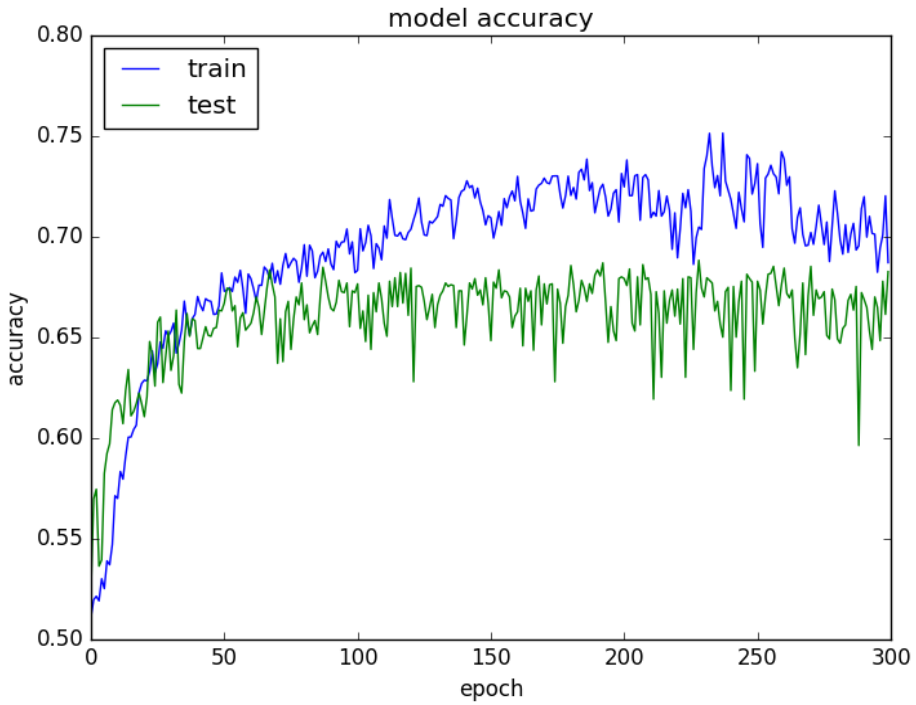

Figure 3.9: The training and testing of system over 300 epochs.

The stored attack is calculated using the key-point descriptors after key-points are extracted from the new attack. Similar performance to KNN with less computational and memory overhead. An illustration of a recent up-right assault, as well as another informational assault

that amplifies the appropriate portion of the assault. Additionally, the areas of the new attack comparable to the reference assault, the eliminated fundamental issues, and the determined matches between examples are displayed. The magnitude is approximately equal to that of the new information onslaught in each clearly discernible area's upright position. Since each location is addressed and warped separately, the cycle can be completed in a comparable amount of time with fewer resources because not the entire assault needs to be handled. For each new district, the preprocessing and RSA operations are carried out. The results are converging with the current ones, and for each location, only the one with the highest RSA certainty is retained. Another attack is requested and carried out as previously described if regions with low recognition scores still exist and the maximum number of retirees has not been reached. The operation halts and the user is alerted of the low score regions if the number of retries is surpassed. The application experiments provide the input samples and RSA output for three attacks, including the fourth attack that was specifically requested. Attack localization, Rivest-Shamir-Adleman, and our method for locating, segmenting, and categorizing cyber security (RSA). This study evaluates each of these processes, with a particular emphasis on classifying cyber security. It also presents experimental findings that were attained utilizing a KNN device. We first assess each feature that was collected from the assault individually. They are examined independently and in combination, with various machine learning algorithm parameters. The result of the RSA is then contrasted with the actual contents of the assault. Only the impact of training with the actual fonts that appear on the assaults is studied because the used RSA is off-the-shelf software with no runtime customization parameters (only training). In the end, we test three potential attack types on a KNN device and provide the findings. This includes various lighting situations and their effects on the processing steps, such as shadows and reflections. Additionally, we demonstrate several breakdown scenarios, such as the attack being captured from unusual angles or blurry images because of inadequate lighting.

Several samples of cyber security from 15 classes make up the dataset utilized for the tests. Because it is difficult to obtain samples of actual attacks from the real world for all the many sorts that are available, the number of samples is rather low. Eleven classes, which are put on deployed devices, are made up of examples of previous attacks. Most of these incidents date back thirty or forty years. The last four classes (k, l, m, and n) were captured using a KNN device and are made up of fresh attacks. All tests are performed using leave-one-out cross-

validation due to the small number of samples, where each sample (from first to last) is classified while the remaining samples are used to train the classifier.

4. RESULTS

This chapter is organized as a series of steps in our method for identifying, categorizing, and extracting assaults from vehicle-related cyber-attacks. To roughly localize the attack and to make a mask for its extraction using KNN, contours are used. Local binary patterns, median color values, and the size ratio make up the feature vector utilized in conjunction with a K-Nearest Neighbors to classify the extracted attack. Attack regions of the retrieved cyber-attack are mapped to the positions of the attack regions that should be extracted for each kind. Prior to being transmitted to the RSA, each region is preprocessed, which enhances the detection outcome. Finally, if the RSA recognition score is poor, the user is asked to perform a new attack on the problematic regions. The suggested actions are assessed in the next section, with particular focus being given to the section on the classification of cyberattacks. There are also some outcomes in various lighting situations and breakdown circumstances provided. The features are first merged pairwise, and the entire feature vector for the classifier is then produced by joining all three together. We display the classification outcomes for each combination. Individual features make up the feature vector, which is calculated for each assault and sent to the K-Nearest Neighbors for classification. The K-Nearest Neighbors algorithm selects m features from the feature vector at each node during training and divides the data using the best split. The association between the layers and the classification strength of each tree are influenced by the number of features m, which is constant throughout the forest. With lowering layer correlation and an increase in the number of powerful classifiers, the classification performance of the forest improves. Additionally, increasing m has the reverse effect of decreasing it, boosting the strength of the individual classifiers and the correlation between them. The square root of the number of variables is a commonly used value, and the research lists the precision and recall values for three typical active variable values when drawing the splits. So from now on, we use this value. The study displays the three attributes' pairwise pairings. As can be observed, the size ratio has the least effect on the outcome whereas the KNN features have the greatest impact. The recall is unchanged, however there is a 0.26 percent precision discrepancy between the final feature vector and just the KNN features. The color information adds relatively little to the classification outcome because the dataset only has a tiny number of samples, which are at most slightly fuzzy. We apply median blurring to the samples in order to replicate the impact of blurred attacks with distorted content.

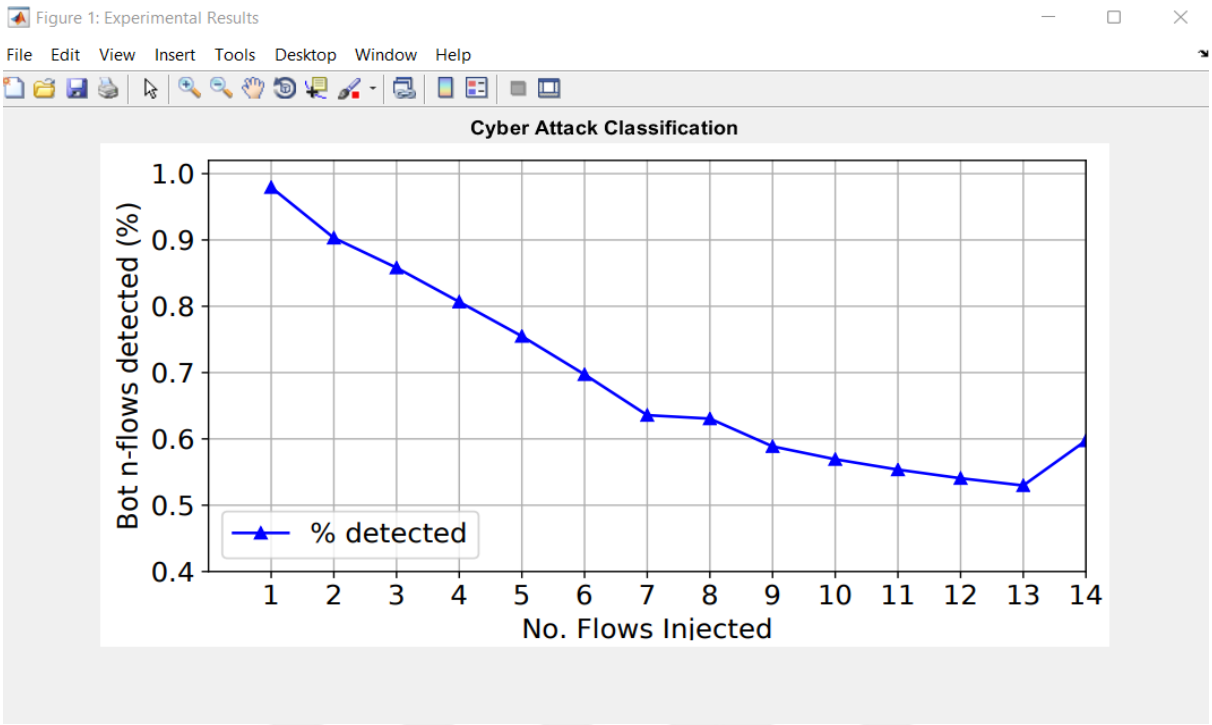

Figure 4.1: Standard normal quantiles represent the predicted vs actual values for cyber security bandwidth ratio with single host.

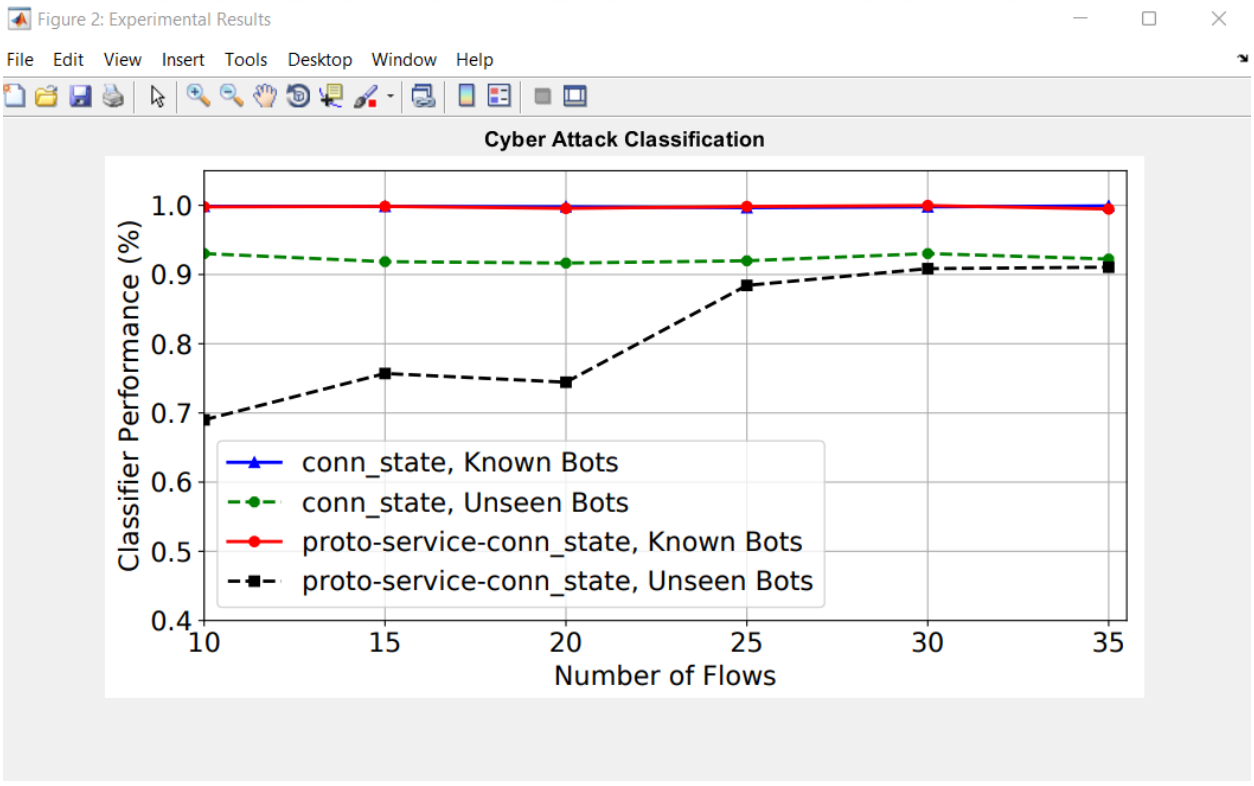

Figure 4.2: The difference between the predicted vs actual values for cyber security with single host.

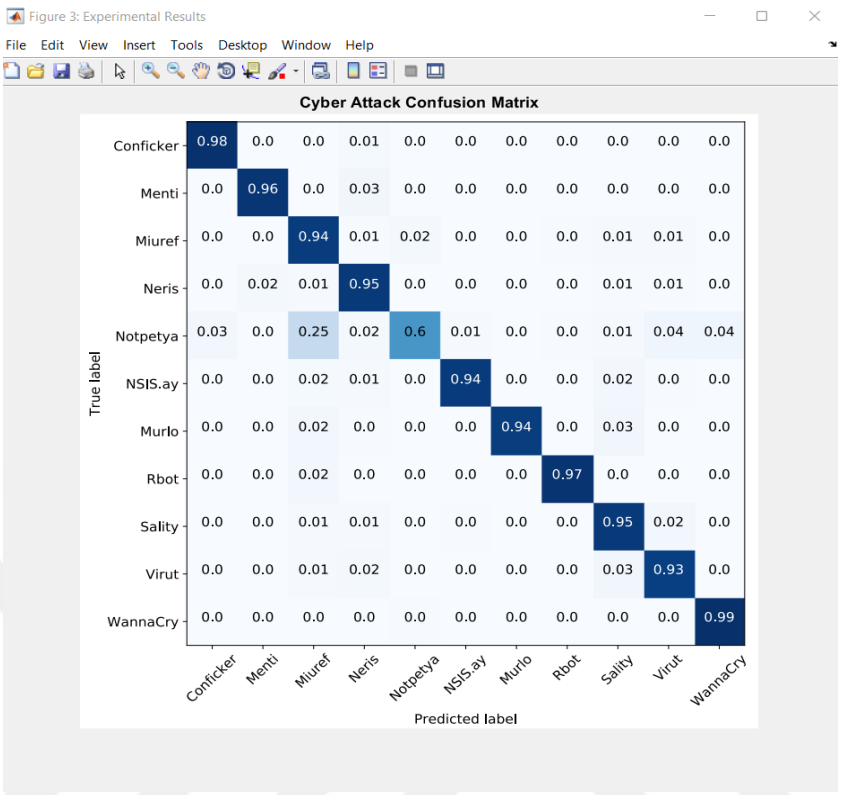

Figure 4.3: Standard normal quantiles represents the predicted vs actual values for cyber security bandwidth ratio with multiple hosts.

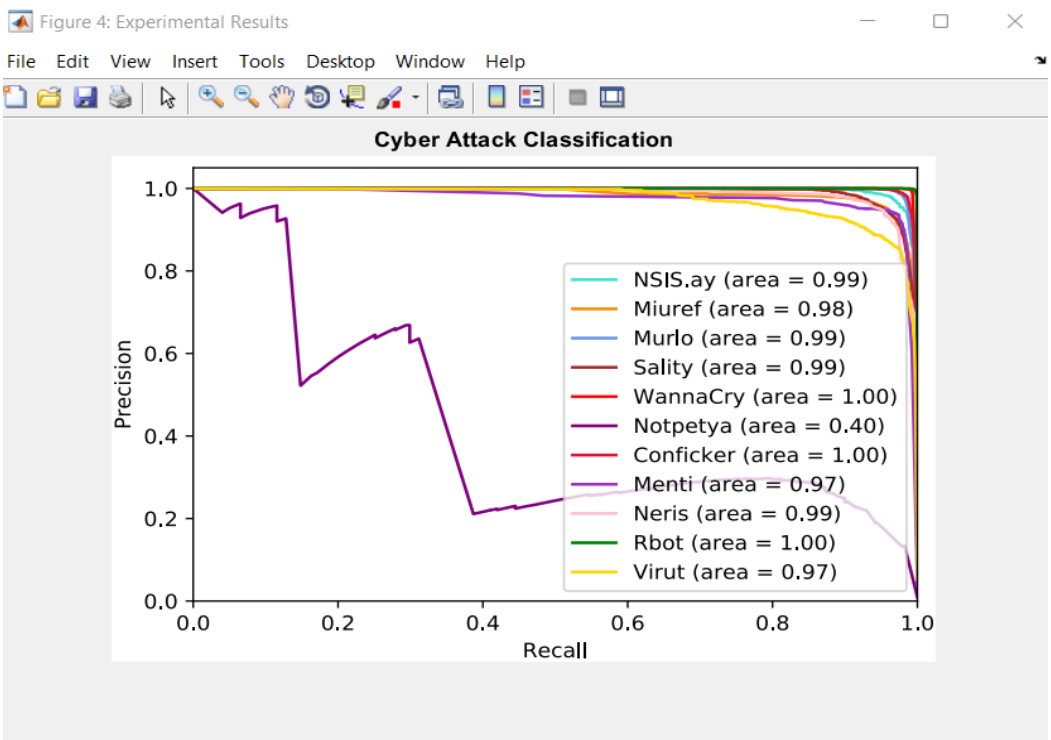

Figure 4.4: The difference between the predicted vs actual values for cyber security with multiple hosts.

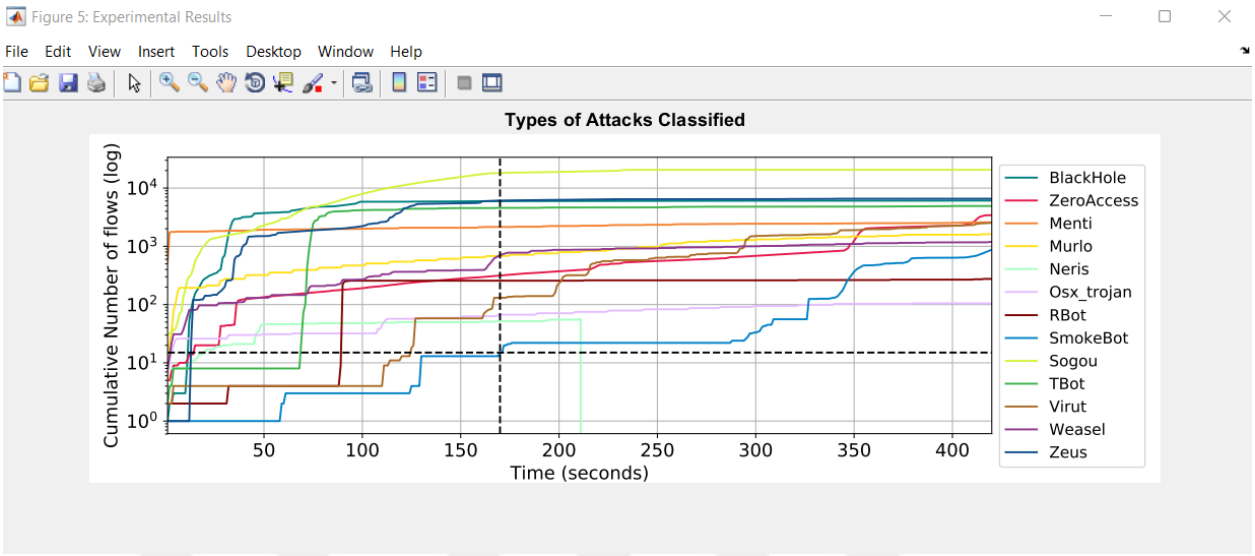

Figure 4.5: Conveyance plot of the recreation time in the simulation for attack detection and predicted types of attacks of node cluster and classification availability bunching calculation for network case in cyber network.

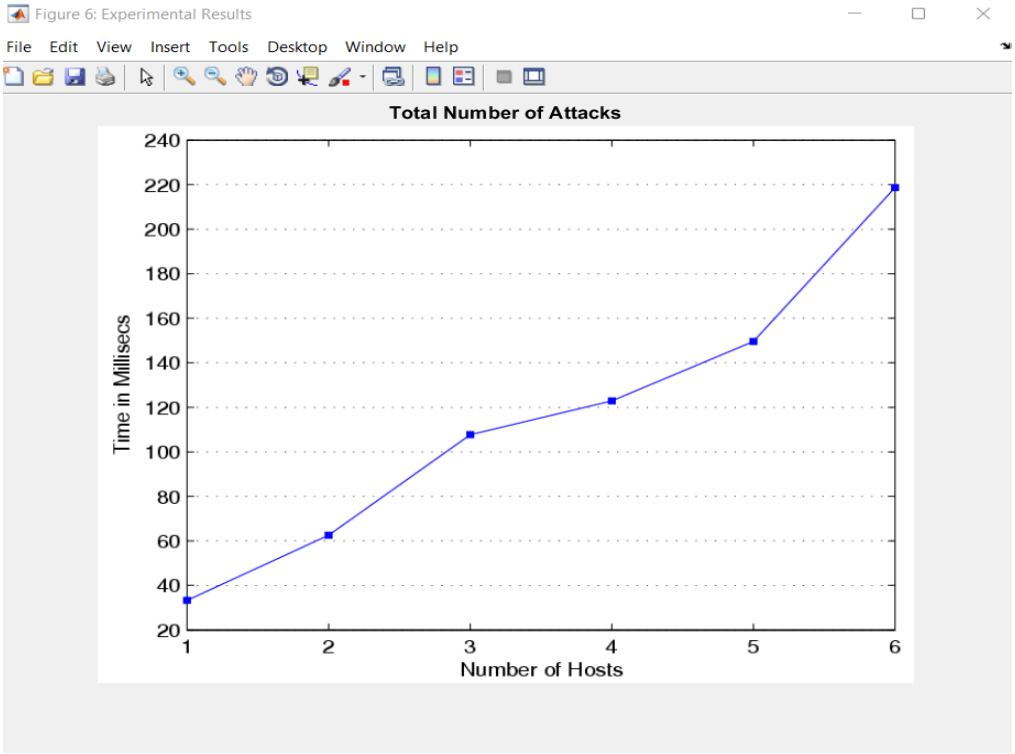

Figure 4.6: Histogram plot of the reproduction time contrasted with the ordinary conveyance for data points in the cyber network.

Typically, the output for the original samples is either incomplete or wrong. The result, on the other hand, is accurate for each preprocessed region. There are, however, instances where the preprocessing altered the original content. Two examples have portions of the cyberspace deleted, while the commas are deleted in the other two examples. It should be observed that the preprocessed output is right in all but one example, whereas the output of the unmodified attack is incorrect. As most of the strings, we need to parse are random combinations of digits and letters, it should be emphasized that we disable the internal dictionary. Consequently, by changing the accurately detected cyberspace, utilizing the dictionary will degrade the outcome. Even with the theoretically low false positive rate of 0.0015, more than 1 in 1000 regular data points would still be incorrectly classified as an assault. A false alert would be issued every 100 seconds if a data point were to be created every 100 milliseconds, which is not practicable. However, considering that a continuous stream of packets is often sent, one data point by itself rarely qualifies as an attack. In that regard, a wireless network strategy might be used, in which an alarm only sounds if most attack data points are contained inside a window of a certain size. The study presents both the KNN features-only classification results and the classification results for the entire dataset with the full feature vector, for various median blur kernel dimensions. We do not use leave-one-out cross validation for this test. Instead, the classifier is initially trained using all the dataset's samples. The feature vector needed for classification is then extracted after blurring every sample. As a result, if no blurring is used, every attack is correctly identified using both evaluated feature vectors.

Table 4.1: The comparison of accuracy of recognition with existing and proposed techniques for cyber security of network.

ARTICLE	TECHNIQUE	ACCURACY
[61]	Support Vector Machine (SVM)	97.69%
[62]	Recurrent Neural Network (RNN)	95.70%
Proposed	K-Nearest Neighbors (KNN)	98.29%

However, we included these findings to demonstrate the classifier's robustness. Additionally, it's feasible that these samples exist in everyday use, for instance, if the camera is out of focus or the

operator moves while recording the attack, causing motion blur. The RSA will fail in these circumstances for every region, but the user will be encouraged to take further samples, which can subsequently be used to produce useful output. After training, K-Nearest Neighbors may determine the significance of each feature in the feature vector. The handbook for the reference K-Nearest Neighbors implementation defines four distinct measures that can be used to determine the variable relevance. The values of the samples not chosen by the bagging process for the tested feature are altered at random and reclassified by the forest. The amount of votes for the correct class in both the original and changed samples are recorded for each tree. The variance, or decline in accuracy, is averaged over all levels and reveals the significance of the examined characteristic. However, the suggested methods are problematic since they could result in false information. Numerous simulation simulations conducted as part of the research demonstrate that the user-selectable number of forest levels has an impact on the variable relevance, but not sample size. Nevertheless, we decided to display the estimated variable importance results. Even if the values are not entirely trustworthy, they nonetheless show which of the three key qualities derived from the attack the forest values the most. The kind of the top 100 features reveals that they are made up of more than 100 KNN values as well as various color values. The size ratio is near the end and has a very low score, which is consistent with our observations when the features are combined pairwise. Additionally, because the forest accuracy is reached before the maximum number of layers are constructed, smaller forests are produced when KNN features are included in the feature vector. 5.50 seconds are required (on average over five runs) to train a forest with all samples. The feature vector must first be retrieved before classification can begin. The attack is scaled to a fixed width, but the height is scaled to maintain the aspect ratio, hence the amount of time required depends on the attack's size ratio.

5. CONCLUSION

This master's thesis developed a way for employing machine learning to identify, categorize, and extract attacks from a known collection of cyber security. The targeted use case featured a user who records attacks connected to cyber security components, such as cropping the attack or producing the region of interest (ROI), using an attack processing technique where the extracted material is processed using a KNN technique based on machine learning. Machine learning is used to segment, localize, and classify the cyber security in the input attack. The attack size ratio, histograms of Local Binary Patterns (LBP), and color information are used as features in the classification stage to determine utilizing KNN. The design of the attack and the locations of the indicated attack sections were different for each type. In order to account for any potential errors generated by the attack extraction, the list of defined fields is compared to the discovered fields on the attack. The noise is removed from the corrected regions before passing them to the Rivest-Shamir-Adleman algorithm (RSA). The RSA is preprocessed to separate related components, size-filter them, and organize them into lines of attack. The well-known KNN approach is the RSA algorithm in use. The user is alerted to acquire a new attack of that portion of the assault, which is then matched to the existing attack, and the attack processing steps are carried out once more if the RSA reported a low confidence score for at least one attack region. The end result includes the type of attack that was identified, a list of the districts that the attack was removed from, and the RSA unwavering quality through KNN with an accuracy of 98.29 percent after 300 iterations of training and testing. The proposed framework was created using 75% of the available data, 20% of which was used for testing, and 5% for validation. Research used a Cyber Data Dataset composed of various classes that addressed a subset of all current types of network safety to illustrate the significance of the intended cyber-attack element vector. It includes both new and mature types. The results revealed that the closest parallel example histograms, followed by the variety highlights, and finally the assault size proportion, contribute more to the order result. Evaluation of the consolidated component vector demonstrated the effectiveness of the arrangement in a variety of lighting scenarios, including the presence of specular reflections and shadows as well as fog and inaccurate variations caused by insufficient illumination. In addition, we showed that the number of recognition failures may be decreased by training the RSA algorithm with the precise typeface of the test assault. The attack region preprocessing,

which eliminates hyphens, commas, and dots when they are small relative to the attack region or blend with its borders, is responsible for the remaining problems. The filtering parameters, which are optimized for older attacks, are to blame for this. Letters that were partially chipped off or faded are likewise removed by the very tight attack region preprocessing. If there is no distinct boundary between the attack and the background, the attack localization may also be wrong or fail. Such situations arise when the attack's borders are overpainted or when dirt and moss build up. If the RSA was educated for the fonts on the cyber security, testing on a KNN with three available assaults produced positive results. Attacks that are highly reflective are challenging to photograph and necessitate manual focus control, which the prototype does not offer. However, given that such attacks degrade quickly when exposed to the elements for an extended period of time, it is unlikely that many of them are mounted to devices.

5.1 FURTHER WORK

We have demonstrated that the suggested KNN method can be applied to the detection of vehicle attacks in real-world scenarios. However, we have some suggestions for how to make it even better. By adding data from the attack samples used to train the deep K-Nearest Neighbors, the segmentation could be improved. Examples include the attack positions or logos, which can be used to estimate the attack border more precisely. Two classes are occasionally misclassified, according to the judgement of the classifier. It is exceedingly challenging to correctly classify the only three visually distinguishable samples in the first class. The second one has a lot of variation as well, mostly in the form of reflections. No modifications to the classification method are required to increase accuracy, although more training samples are required. The preprocessing and DES algorithm are the final phases that could use some enhancements. It is exceedingly unlikely to always produce flawless results because some assaults—particularly embossed ones and those with very ancient content—are challenging for even humans to identify. One option would be to individually preprocess various marking modalities. For instance, an embossed assault is considerably fainter, but the metal around the letters is rather noisy due to the imprinting process. Additionally, the identified material, such as decimal separators of digits or hyphens in serial numbers, might be automatically verified and corrected with the help of knowledge about the format of the field contents. Despite being the greatest open-source solution currently available, the utilized DES is not optimized for our use case. Therefore, creating a new DES that is resistant to noise would be yet another option.

REFERENCES

[1] S. Du, M. Ibrahim, M. Shehata and W. Badawy, “Automatic Cyber Security Recognition (ALPR): a state-of-the-art review,” IEEE Transactions on Circuits and Systems for Video Technology 23(2), pp. 311-325, 2019.

[2] Sayanan Sivaraman and Mohan M. Trivedi, "A General Active Learning Framework for On-road Vehicle Recognition and Tracking," IEEE Transactions on Intelligent Transportation Systems, 2020.

[3] S. Silva and C. Jung, “Real-time Brazilian cyber security detection and recognition using deep K-Nearest Neighbors,” in 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), 2017.

[4] Matas, Jiri & Zimmermann, Karel. (2015). Unconstrained cyber-attack and cyber-attack localization and recognition. 225 - 230. 10.1109/ITSC.2015.1520111.

[5] Ansari, N.N.; Singh, A.K.; Student, M.T. Cyber-attack Number Attack Recognition using Temattack Matching. Int. J. Comput. Trends Technol. 2016, 35, 175–178.

[6] 12. Samma, H.; Lim, C.P.; Saleh, J.M.; Suandi, S.A. A memetic-based fuzzy K-Nearest Neighbors model and its application to cyber security recognition. Memetic Comput. 2016, 8, 235–251.

[7] R. Girshick, “Fast R-ANN,” in IEEE International Conference on Computer Vision (ICCV), 2015.

[8] . Ren, K. He, R. Girshick and J. Sun, “Faster R-ANN: towards real-time vehicle detection with region proposal networks,” in Conference on Neural Information Processing Systems (NIPS), 2015.

[9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu and A. Berg, “SSD: single shot multibox detector,” in The European Conference on Computer Vision (ECCV), 2016.

[10] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi and A. C. Berg, “Dssd: deconvolutional single shot detector,” arXiv preprint arXiv:1701.06659, 2017.

[11] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan and C. L. Z. P. Dollár, “Microsoft COCO: Common Network in Context,” in The European Conference on Computer Vision (ECCV), 2016.

[12] Rizvi, S.; Patti, D.; Björklund, T.; Cabodi, G.; Francini, G. Deep Classifiers-Based Vehicle Attack Detection, Localization and Recognition on GPU-Powered Mobile Platform. Future Internet 2017, 9, 66.

[13] Rafique, M.A.; Pedrycz, W.; Jeon, M. Cyber security detection using region-based K-Nearest Neighbors. Soft Comput. 2018, 22, 6429–6440.

[14] Salau, A.O.; Yesufu, T.K.; Ogundare, B.S. Vehicle attack number localization using a modified GrabCut algorithm. J. King Saud Univ. Comput. Inf. Sci. 2019.

[15] Kakani, B.V.; Gandhi, D.; Jani, S. Improved RSA based automatic vehicle number attack recognition using features trained neural network. In Proceedings of the 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Delhi, India, 3–5 July 2017.

[16] Arafat, M.Y.; Khairuddin, A.S.M.; Paramesran, R. A Vehicular Cyber Security Recognition Framework for Skewed Samples. KSII Trans. Internet Inf. Syst. 2018, 12.

[17] Available Online: https://melabglobal.com/blogs/news/why-use-cyber attack-attack-recognition-cyber-system

[18] Tabrizi, S.S.; Cavus, N. A Hybrid KNN-KNN Model for Iranian Cyber Security Recognition. Procedia Comput. Sci. 2016, 102, 588–594.

[19] Available Online: https://medium.datadriveninvestor.com/convolutional-neural-networks-explained-7fafea4de9c9

[20] Niu, B.; Huang, L.; Hu, J. Hybrid Method for Cyber Security Detection from Natural Scene Samples. In Proceedings of the First International Conference on Information Science and Electronic Technology, Wuhan, China, 21–22 March 2015; Atlantis Press: Paris, France, 2015.

[21] Arafat, M.Y.; Khairuddin, A.S.M.; Khairuddin, U.; Paramesran, R. Systematic review on vehicular licence attack recognition framework in intelligent transport systems. IET Intell. Transp. Syst. 2019.

[22] Chai, D.; Zuo, Y. Extraction, Segmentation and Recognition of Vehicle's Cyber Security Numbers. In Advances in Information and Communication Networks; Arai, K., Kapoor, S., Bhatia, R., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 724–732.

[23] Dhar, P.; Guha, S.; Biswas, T.; Abedin, M.Z. A System Design for Cyber Security

Recognition by Using Edge Detection and Convolution Neural Network. In Proceedings of the 2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2), Rajshahi, Bangladesh, 8–9 February 2018; pp. 1–4.

[24] Wang, W. et al. "Car cyber security detection based on MSER." 2011 International Conference on Consumer Electronics, Communications and Networks (CECNet) (2011): 3973-3976.

[25] Brillantes, A.K.M.; Bandala, A.A.; Dadios, E.P.; Jose, J.A. Detection of Fonts and Cyberspace with Hybrid Graphic-Attack Attack Numbers. In Proceedings of the TENCON 2018—2018 IEEE Region 10 Conference, Jeju, South Korea, 28–31 October 2018; pp. 629–633.

[26] J. R. Uijlings, K. E. v. d. Sande, T. Gevers and A. W. Smeulders, "Selective search for vehicle recognition," International Journal of Computer Vision (IJCV), 2013.

[27] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus and Y. LeCun, "OverFeat: integrated recognition, localization and detection using convolutional networks," arXiv preprint arXiv:1312.6229, 2013.

[28] J. Redmon and A. Farhadi, "YOLOv3: an incremental improvement," arXiv preprint arXiv:1804.02767, 2018.

[29] T.-Y. Lin, P. Goyal, R. Girshick, K. He and P. Doll´ar, "Focal loss for dense vehicle detection," in IEEE International Conference on Computer Vision (ICCV), 2017.

[30] H. Law and J. Deng, "CornerNet: detecting network as paired keypoints," in The European Conference on Computer Vision (ECCV), 2018.

[31] C. Zhu, Y. He and M. Savvides, "Feature selective anchor-free module for single-shot vehicle detection," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[32] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale attack recognition," in International Conference on Learning Representations (ICLR), 2015.

[33] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for attack recognition," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[34] A. Newell, K. Yang and J. Deng, "Stacked hourglass networks for human pose

estimation," in The European Conference on Computer Vision (ECCV), 2016.

[35] O. Ronneberger, P. Fischer and T. Brox, "U-net: convolutional networks for biomedical attack segmentation," in International Conference on Medical attack computing and computer-assisted intervention, 2015.

[36] Y. Lin, P. Doll´ar, R. Girshick, K. He, B. Hariharan and S. Belongie, "Feature pyramid networks for vehicle detection," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[37] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[38] S.-H. Kuo, J.-E. Ok and E.-Y. Cha, "Vehicle Front-Rear detection for cyber security detection enhancement," in 15th International Conference on Multimedia Information Technology and Applications(MITA), 2019.

[39] Available Online: https://www.chegg.com/homework-help/questions-and-answers/hello-m-trying-detect-driving-lanes-using-hough-transform-matlab-think-ve-successful-excep-q28180023

[40] Available Online: https://gitmemory.com/issue/AlexeyAB/darknet/1492/480806996

[41] A. Shrivastava, A. Gupta and R. Girshick, "Training region-based vehicle detectors with online hard example mining," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[42] Z. Xu, W. Yang, A. Meng, N. Lu and H. Huang, "Towards end-to-end cyber security detection and recognition: A Large dataset and baseline," in The European Conference on Computer Vision (ECCV), 2018.

[43] 18. Zhu, Z.; Ren, X.; Chen, Z. Integrated detection and tracking of workforce and equipment from construction jobsite videos. Autom. Constr. 2017, 81, 161–171.

[44] Rezazadeh Azar, E.; McCabe, B. Part based model and spatial–temporal reasoning to recognize hydraulic excavators in construction samples and videos. Autom. Constr. 2012, 24, 194–202.

[45] Kim, J.; Chi, S. Action recognition of earthmoving excavators based on sequential pattern analysis of visual features and operation cycles. Autom. Constr. 2019, 104, 255–264.

[46] Akhavian, R.; Behzadan, A.H. Construction equipment activity recognition for simulation input modeling using mobile sensors and machine learning classifiers. Adv.

Eng. Inform. 2015, 29, 867–877.

[47] Kim, J.; Chi, S.; Seo, J. Interaction analysis for vision-based activity identification of earthmoving excavators and dump trucks. Autom. Constr. 2018, 87, 297–308.

[48] Gong, J.; Caldas Carlos, H. Computer Vision-Based Video Interpretation Model for Automated Productivity Analysis of Construction Operations. J. Comput. Civ. Eng. 2010, 24, 252–263.

[49] Silva, S.M.; Jung, C.R. Real-Time Cyber Security Detection and Recognition Using Deep K-Nearest Neighbors. J. Vis. Commun. Attack Represent. 2020, 71, 102773.

[50] Omar, N.; Sengur, A.; Al-Ali, S.G.S. Cascaded machine learning-based efficient approach for cyber security detection and recognition. Expert Syst. Appl. 2020, 149, 113280.

[51] Wang, D.; Tian, Y.; Geng, W.; Zhao, L.; Gong, C. LPR-Net: Recognizing Chinese cyber security in complex environments. Pattern Recognit. Lett. 2020, 130, 148–156.

[52] Hsu, G.; Chen, J.; Chung, Y. Application-Oriented Cyber Security Recognition. IEEE Trans. Veh. Technol. 2013, 62, 552–561.

[53] Masood, S.Z.; Shu, G.; Dehghan, A.; Ortiz, E.G. Cyber security detection and recognition using deeply learned K-Nearest Neighbors. arXiv 2017, arXiv:1703.07330.

[54] Saraswathi, S.; Subban, R.; Shanmugasundari, T.; Manogari, S. Research on Cyber Security Recognition Using Digital Attack Processing. In Proceedings of the 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Coimbatore, India, 14–16 December 2017; pp. 1–6.

[55] Khan, A.M.; Awan, S.M.; Arif, M.; Mahmood, Z.; Khan, G.Z. A Robust Segmentation Free Cyber Security Recognition Method. In Proceedings of the 2019 International Conference on Electrical, Communication, and Computer Engineering (ICECCE), Swat, Pakistan, 24–25 July 2019; pp. 1–6.

[56] Liu, L.; Ouyang, W.; Wang, X.; Fieguth, P.; Chen, J.; Liu, X.; Pietikäinen, M. Machine learning for Generic Object Detection: A Survey. Int. J. Comput. Vis. 2020, 128, 261–318.

[57] Puarungroj, W.; Boonsirisumpun, N. Thai Cyber Security Recognition Based on Machine learning. Procedia Comput. Sci. 2018, 135, 214–221.

[58] Cao, Y.; Fu, H.; Ma, H. An End-to-End Neural Network for Multi-Line Cyber Security

Recognition. In Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 3698–3703.

[59] Chen, R.C. Automatic Cyber Security Recognition via sliding-window darknet-YOLO machine learning. Attack Vis. Comput. 2019, 87, 47 – 56.

[60] Available Online: https://medium.com/@anand.mishra/using-segmentation-analysis-to-improve-marketing-10b4a95bf62c

[61] Rizvi, Murtaza & Singh, Shailendra & Agrawal, Sanjay & Thakur, R. (2011). Improved Support Vector Machine for Cyber-Attack Detection.

[62] Usmankhujaev, Saidrasul & Lee, Seonwoo & Kwon, Jangwoo. (2020). Korean Cyber Security Recognition System Using Combined Neural Networks. 10.1007/978-3-030-23887-2.