



## **Network Security**

### Section

## **ANALYSIS FOR VARIOUS DIGITAL FORENSIC AND THEIR OPEN SOURCE DIGITAL FORENSIC TOOLS WITH EFFECTIVE FUNCTION IN CRIME INVESTIGATION**

**Ohn Mar Myint <sup>(1)</sup>, Moe Moe Thein <sup>(2)</sup>, Nyein Nyein Hling <sup>(3)</sup>**

<sup>(1)(2)(3)</sup>University of Computer Studies (Meiktila), Myanmar

<sup>(1)</sup> *0hnmar1311@gmail.com*

### **ABSTRACT**

The world has been dramatically changed with the evolution of technology, widely used computer, mobile phone and etc., Technology becomes an important role in human life nowadays and it also provides immense opportunities. The access to education, medicine, transportation etc. has been simplified due to modern day technology. Beside these advantages, there are also some disadvantages like cyber-crime by using Internet, Mobile phones and digital devices. In this paper, we will introduce current various kinds of Digital forensics and their respecting digital forensics tools to investigate digital crimes more efficiently and effectively. These are shown by table.

**KEYWORDS:** *Forensic, forensic tools.*

### **1. INTRODUCTION**

In the context of computer and Internet related crimes, Computer forensics is also included in the main importance of computer science. The main objective of Computer forensics is the use of evidence from digital data to find who was the responsible for that particular crime for the investigations.

Digital forensics may be defined as the application of examination and analysis techniques to gather and preserve evidence from an appropriate computing device in a suitable way for presentation in a court of law.

Computer forensics is defined as “the application of computer investigation and analysis techniques in the interests of determining potential legal evidence” [1].

Acquiring, authenticating, and analysis of the data are the three major steps of computer forensic analysis. The data collection mainly includes creating a bit-by-bit copy of the hard drive. Authentication is the process or action of proving that the copy used to perform the analysis is an exact copy of the contents of the original hard drive by comparing the checksums of the copy and the original.[2]

Forensics methodologies and forensics tools are very importance of any digital forensics. By choosing proper Digital Forensics tools, computer analysis and digital evidence could be gotten. With them, we can investigate computer crimes, making them into strong evidences and sending to the court of law.

These computer forensics tools can also be classified into various categories:

- 1.Disk and data capture tools
- 2.File viewers
- 3.File analysis tools
- 4.Registry analysis tools
- 5.Internet analysis tools
- 6.Email analysis tools
- 7.Mobile devices analysis tools
- 8.Mac OS analysis tools
- 9.Network forensics tools
- 10.Database forensics tools

The rest of the paper is organized as follows:  
Section 2 Digital Forensics Methodology Section 3

describes Digital Forensics investigation topic. Detail description of forensic tools in 4. Section 5 compose on findings from the analysis. We present conclusion in last section 6.

## **2. DIGITAL FORENSICS METHODOLOGY**

Methodologies are constituted for assisting the proper series of actions concluded in an investigation. Some of the methodologies are brief and can be used in any situation which concerns digital evidence and others are aimed at a certain tool.

Digital forensic methodology is to investigate a malicious web page and malicious digital devise. Clear, exact methodologies and plans for suitable situations can be defined by digital forensics examiners.

During the computer forensics investigation there are a variety of steps that must be taken. The following step, defined in the book Computer forensics: Incidence Response. Each of these steps can be further refined.

1. Acquire the Evidence
2. Authenticate the Evidence
3. Analysis the Evidence
4. Present the Evidence

The brief model consists of nine phrases-identification, preparation, approach strategy, preservation, collection, examination, analysis, presentation, and returning evidence. The benefit of the succinct model is that we can use it in any situation where digital evidence has been comprised, not only for examining computers. [3]

## **3. DIGITAL FORENSICS INVESTIGATION TOPICS**

The most stated articles grouped on journal and online published, dealing with Digital Forensics investigation are

### **i. Digital Image Forensics**

The usage of digital images had increased wildly. They came to involve in crimes such as image forgery which are increased subsequently and many difficulties to forensic investigators and researchers which were brought by the detection of the image tempering. The main detectable research topics in digital image forensics are image authenticity, image

correction, steganography, and image processing and source detection.

### **ii. Mobile Device Forensics**

Due to the wild increase in usage of mobile devices such as smart phones, tablets and GPS devices, investigation of such devices is apparently essential and important. Hence it is a great value to obtain and analyze the evidence from mobile devices. Further the major detectable research topics are Mobile live memory, Mobile forensic framework, Mobile forensic tools and Mobile on-scene triage topics.

### **iii. Network Forensics**

Apart from providing the unlimited access to internet application and ability to communicate between each other, the reliability of other IT gadget networking lies on how they can be effective in corresponding to another device. Lower layer binary network signature like socket and packet data structure is depended on by the communication among those implements. For investigators, it is a great challenge to obtain evidences from network traffics due to the live characteristics of network packets.

### **iv. Memory Forensic**

While the computer is seized, memory forensics examines the information captured from memory. Since the focus has been less paid to extracting information from Window drivers, they are developing a methodology for minimizing the effort of analyzing these drivers. While modern Windows operating systems cache file data in memory aggressively, current forensic tools and techniques do not take mapped-file information into account.

### **v. Volatile Memory Forensics**

As the basic requirements of knowing the structure of memory is not satisfied, another challenge for forensic investigators has extracted potential evidences from volatile memories. RAM analysis is concerned with the repossession of information, like evidence in analysis of crime. More specifically, memory management structures in computer map the abstracted files and executables resident in a computer's physical memory [4].

#### **vi. Database forensics**

The necessity of company or organization to do business is the secure data storage. For many companies or organizations, it could cause consequences for the company if data was altered by an outsider or an inside intruder. Database systems make numerous plenty of copies of sensitive data, so when it is deleted, it is not destroyed entirely but it often presents on disk. It becomes increasingly important because of the information relating to various inquiries in a database. Database often contains information that helps in solving a crime but it may not focus on the crime.

#### **vii. Application System Forensics**

Storing specific evidences by the forensic investigation of application is quite useful. Prior research on the application behavior is demanded for by identifying and collecting these evidences. This study both demonstrated approaches and told to detect different attributes of download requests like URL, download time and login credentials.

#### **viii. File Forensic**

Until the last cluster of the file, the route table entry with the file name will point to the first cluster of the file, which in turn will point to the next cluster and so on in the file system FAT\_32. By the time a file is deleted, the actual content located in several clusters in the storage device is not removed from the table but only from the file's entry. The file system declares the unallocated space from which the file will be recovered. The main problem is to recover a file in digital forensic, file carver which fails to recover a fragmented file.

#### **ix. Email Forensics**

The fastest method to recover the data or email lost in the digital forensic investigation is by using the reverse engineering from how the email is deleted. As the file can be very fragile, the evidence of lost file in email said to be handled with care. It is important of establishing the authenticity of an electric file or email in the organization. Within temporary files, replicated, swap files, other system-created files or in a computer's unallocated space, the evidence is buried. An incident handler is for discovering the buried evidence. The task is accomplished through storage media searches relating to previous deleted or erased documents, parts of documents or drafts of documents. Parts of

the document may include private data, not concerning with the crime.

#### **x. Operating system Forensics**

System Forensics is the process of retrieving useful information from the Operating System (OS) of the computer or mobile device in question to obtain realistic evidence against the perpetrator. Forensics O.S. Forensics is a toolkit that provides lots of information about the use of a computer and the files stored in it. In OS Forensics you can easily manage your tasks and programs. OS forensics is a complete selection of tools and its interface is neatly organized. The program is installed directly on a memory device and a USB device.

#### **xi. Removable Media Forensic**

To give you a list of all USB drives that were plugged into the machine, USB information is analyzed from the window registry by USB historian. It displays information such as the name of the USB drive, the serial number, when it was mounted and by which user account. This information can be very useful when you're dealing with an investigation whereby you need to understand if data was stolen, moved or accessed.

#### **xii. Website Forensics**

To perform different activities on the internet such as browsing internet, email, social media applications and so on, internet users use the web browser which is the only way to access the internet and cybercrime criminal uses or target the web browser to commit the internet crime. Collecting and analyzing artifacts related to web browser usage of the suspect is very important for the digital forensic examiner.

### **4. DIGITAL FORENSICS TOOLS**

Two kinds of Digital Forensics tools are—hardware and software. They are used in the process of recovery and preservation of digital evidence.

Law enforcement uses digital forensics software and hardware interchangeably. Most products are available to law enforcement, whether open source or commercial; concentrate on computer and mobile device forensics, as these two branches are more prevalent

Hardware tools are designed primarily for storage device investigations, and they intend to keep

suspect devices unaltered to preserve the integrity of evidence. A forensic disk controller or a hardware write- blocker is a read-only device that allows the user to read the data in a suspect device without the risk of modifying or erasing the content.

#### **i. X-Ways Forensics:- Integrated computer forensic software**

X-Ways forensic, an advanced work environment for digital forensic examiners, is more efficient to use after a while, often run faster, is not as resource-hungry, finds deleted files. X-Ways forensics is fully portable, runs off a USB stick on any given windows system without installation. It is based on the WinHex hex and Disk editor and part of an efficient work flow model where computer forensic examiners share data and collaborate with investigators that use X-Ways investigator [5].

#### **ii. SANS**

Investigative Forensic Toolkit-SIFT is a multi-purpose forensic operating system which comes with all the necessary tools used in the digital forensic process [6].

#### **iii. Volatility**

Volatility is the memory forensics framework. It used for incident response and malware analysis. With this tool, you can extract information from running processes, network sockets, network connection, DLLs and registry hives. It also has support for extracting information from Windows crash dump files and hibernation files. This tool is available for free under GPL license. Read more about the ol [7].

#### **iv. The Coroners's Toolkit**

The coroner's Toolkit or TCT is also a good digital forensic analysis tool. It runs under several UNIX\_ related OS. TCT is a collection of programs by Dan Sarmer and Wietse Venema for a post-mortem analysis of a UNIX system. The software was presented first in computer forensics analysis class in August 1999 [8].

#### **v. Cofee**

Computer Online Forensic Evidence Extractor or COFEE is a tool kit developed for computer forensic experts. This tool was developed by Microsoft to gather evidence from windows system. Just plug in the USB device in the target computer

and it starts a live analysis. It is fast and can perform the whole analysis in as few 20 minutes. To law enforcement agencies Microsoft provides free technical support for the tool [9].

#### **vi. Bulk Extractor**

Bulk Extractor is also an important and popular digital forensic tool. It scans the disk images, file or directory of files to extract useful information. In this process it ignores the file system structure so it is faster than other available similar kinds of tools. It is basically used by intelligence and law enforcement agencies in solving cybercrimes [10].

#### **vii. Caine**

CAINE(Computer Aided Investigative Environment) is the Linux distro created for digital forensics. It offers an environment to integrate existing software tools as software modules in a user friendly manner. This tool is open source [11].

#### **viii. Forensics Toolkit (FTK) Imager**

FKT imager is a data preview and imaging tool that allow examine files and folders on local hard drives; network drives, CDs/DVDs, and review the content of forensic images or memory dumps. Using FTK Imager also create SHA1 or MD5 hashes of files, export files and folders from forensic images to disk, review and recover files that were deleted from the Recycle Bin, and mount a forensic image to view its contents in Windows Explorer.

#### **ix. Browser History Examiner**

Browser History Examiner extracts, analyzes web history and supports Chrome, Firefox, Internet Explorer and Edge web browsers. It can analyze a lot of data type as downloads, cache data and visited URL files. Internet activities in a specific timeline can be traced with web site timeline feature. Data can be analyzed by using various filters like key word list and time-date range with advanced filtering feature. Image files saved in the browser cache can be easily shown in thumbnail galleries with cache image viewer feature.

#### **x. P2P tool**

The most useful Peer to Peer file sharing applications and program are a. Bittorrent. This is one of best, fastest, and widely, used P2P program. Performance of bit torrent is way better than any other available P2P program. Support operating system windows, Linux and Mac.

**xi. Autopsy**

Autopsy is a digital forensics platform. It based on GUI program that allows you to investigate hard disk drives, in-depth analysis of various file system and smart phones. It has a plug-in architecture that allows you to find add-on programs or develop custom programs in Java or Python. Autopsy runs as a web server, and can be accessed using an HTML browser. Installation is easy and wizards guide you through every step. All results are found in a single tree.

**xii. DFF (Digital Forensics Framework)**

It is an open source tool which will help you in your digital forensics works, including files restoration owing to error or crash, evidence research and analysis etc. It is used by not only professional but also non- expert people to gather quickly and easily, conserve and admit digital evidences without compromising systems and data. Digital Forensic Framework can be installed on Linux and Windows.

**xiii. Wireshark**

Wire shark, a free open-source packet analyzer, used for network damage, investigation, software and connection protocol advancement, and education. We can observe all packets in network and recognize high level of traffic in our network through it. Wire shark is available for all OS and GUI environment which provide user friendly interface. It runs on UNIX-based systems, Mac OS X, and windows.

**xiv. FAW (Forensics Acquisition of Websites)**

This is the first browser that can acquire web pages from websites available online to conduct forensic investigation.

Its key features include: Viewing and editing host files. Audio/video capture. Acquiring code for iFrames on the webpage. Acquiring IP address and hostname of webpage. Support for English, French, Italian, and Polish languages. Improved performance and stability. Pros: It extracts image files on webpages being viewed. It can capture files such as JavaScript and CSS on a website, which can help detect malware. It preserves a webpage while it is being viewed by a user.

**xv. USB Historian**

This tool can both analyze all your USB history information from your windows plug-and-play

registry and give you a complete record of the USB drives that were inserted into the machine. The tool is originally intended to conduct forensic investigations related to stealing, movement, or unauthorized access to data.

**xvi. Fire Eye Redline**

RedLine offers the ability to perform memory and file analysis of a specific host. It collects information about running processes and drivers from memory, and gathers file system metadata, registry data, event logs, network information, services, tasks, and Internet history to help build an overall threat assessment profile. As soon as you launch RedLine, you will be given a choice to Collect Data or Analyze Data. When you have a memory dump file to hand, you can begin your analysis. [12]

**xvii. Xplico**

Xplico is an open source Network Forensic Analysis Tool (NFAT) that aims to extract applications data from internet traffic. Features include support for a multitude of protocols (e.g. HTTP, SIP, IMAP, TCP, UDP), TCP reassembly, and the ability to output data to a MySQL or SQLite database, amongst others. Once you've installed Xplico, access the web interface by navigating to 9876(IP) and logging in with a normal user account.

**xviii. PlainSight**

PlainSight is a Live CD based on Knoppix (a Linux distribution) that allows you to perform digital forensic tasks such as viewing internet histories, data carving, examining physical memory dumps, extracting password hashes, and more. When you boot into PlainSight, a window pops up asking you to select whether you want to perform a scan, load a file or run the wizard [13].

**xix. Helix3 Free**

HELIX3 is a Live CD based on Linux that was built to be used in Incident Response, Computer Forensics and E-Discovery scenarios. It is packed with a bunch of open source tools ranging from hex editors to data carving software to password cracking utilities, and more. When you boot using HELIX3, you are asked whether you want to load the GUI environment or install HELIX3 to disk. [14]

**xx. ExifTool**

ExifTool, a command-line application used to read, write or edit file metadata information is fast,

powerful and supports a large range of formats. ExifTool can be used to analyze the static properties of suspicious files in a host-based forensic investigation, for example. To use ExifTool, simply and drop the file you want to extract metadata from onto the exifTool.exe application and it will open a command prompt window with the information displayed. Alternatively, rename exiftool (-k).ex to exifTool.exe and run from the command prompt [12].

## 5. ANALYSIS RESULTS

This is we own analysis result for forensics researchers of developing country

no	Digital Forensics investigation topic	Digital Forensics Tools															
		X-Ways forensics	SANS	Volatility	The Coroner's Toolkit	COFFE	BULK EXTRACTOR	CARE	FTK Imager	unwired memory	Examine	P2P tool	digital forensics Framework	Autopsy	WIRESHARK	PAW	USB Historian
1	Digital Image Forensics						✓		✓			✓					
2	Mobile Device Forensics							✓						✓			
3	Network Forensics					✓		✓				✓			✓		✓
4	Memory Forensics		✓					✓									✓
5	Volatile Memory Forensics			✓									✓				✓
6	Database Forensics													✓			
7	Application System Forensics																
8	File Forensics	✓	✓	✓			✓							✓			✓
9	Email Forensics					✓	✓										
10	Operating system				✓												✓
11	Removable Media Forensic	✓														✓	
12	Website Forensics									✓		✓	✓	✓	✓		

## 6. CONCLUSION

Methodologies assist to offer the proper series of actions needed for an investigation. For any situation concerning with digital evidence, some methodologies are brief and other give a proper tool.

Latest, popular, open source digital forensics tools are used by various law enforcement agencies in performing crime investigations. These tools are designed with multiple purposes-intending to precise situations for various kinds of digital forensics. In this analysis, the current different kinds of Digital

forensics and the 2019 popular, open source digital forensics tools helps the investigation of digital crimes more efficiently and effectively. Open source Digital forensics tools assist for researchers of developing country because most Digital forensic tools are expensive.

## REFERENCES

- [1] "References Al Zarouni Marwan Mobile Handset Forensic Evidence A Challenge for."
- [2] M. Kaur, N. Kaur, and S. Khurana, "A Literature review on Cyber Forensic and its Analysis

- tools,” *IJARCCCE*, vol. 5, no. 1, pp. 23–28, Jan. 2016.
- [3] N. Balon, R. Stovall, and T. Scaria, “Computer Intrusion Forensics Research Paper,” p. 24.
- [4] O. M. Adedayo and M. S. Olivier, “Reconstruction in Database Forensics,” in *IFIP Int. Conf. Digital Forensics*, 2012.
- [5] “X-Ways Forensics: Integrated Computer Forensics Software
- [6] “Digital Forensics Training | Incident Response Training | SANS.”
- [7] “Google Code Archive - Long-term storage for Google Code Project Hosting.”
- [8] “ComputerForensicAnalysis.” <http://www.porcupine.org/forensics/>. [Accessed: 13-Dec-2019].
- [9] “cofee.nw3c.”: coffe.nw3c.org.
- [10] Y. Yannikos, L. Graner, M. Steinebach, and C. Winter, “Data Corpora for Digital Forensics Education and Research,” in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, vol. 8827, E. Bayro-Corrochano and E. Hancock, Eds. Cham: Springer International Publishing, 2014, pp. 309–325.
- [11] “CAINE Live USB/DVD - computer forensics digital forensics.”
- [12] “United States Department of Homeland Security,” 04-Dec-2019.
- [13] W. Jansen and R. Ayers, “An Overview and Analysis of PDA Forensic Tools,” *Digit. Investig.*, vol. 2, pp. 120–132, Jun. 2005.
- [14] “Top 20 Free Digital Forensic Investigation Tools for SysAdmins - 2019 update,” *GFI Blog*, 11-Sep-2019.
- [15] Preeti, Sushil, *Tool and Techniques for Computer Forensics*. .
- [16] “A Guide to Digital Forensics and Cybersecurity Tools,” *Forensics Colleges*.
- [17] “Ministry of Communications and Information Technology (Egypt),” *Wikipedia*. 07-Nov-2019.
- [18] “Computer Forensics: Incident Response Essentials [Book].”
- [19] J. Oh, S. Lee, and S. Lee, “Advanced evidence collection and analysis of web browser activity,” *Digit. Investig.*, vol. 8, pp. S62–S70, Aug. 2011.
- [20] K. P. Chow and S. Yiu, “Tools and Technology for Computer Forensics: Research and Development in Hong Kong (Invited Paper),” 2007, vol. 4464, pp. 11–19.
- [21] Department of Digital Forensics Engineering, Fırat University Technology Faculty, 23119, Elazığ, Turkey., E. Akbal, F. Güneş, and A. Akbal, “Digital Forensic Analyses of Web Browser Records,” *J. Softw.*, vol. 11, no. 7, pp. 631–637, Jul. 2016.
- [22] *International Journal of Computer Sciences and Engineering*. Vol. 5, Issue 1, January 2016.

## **DEVELOPMENT OF HIGH DATA CAPACITY COLORED QR CODE**

**Myo Min Hein<sup>(1)</sup>**

<sup>(1)</sup>Department of Computer Science, DSA, Pyin Oo Lwin, Myanmar

<sup>(1)</sup>*myominhein48@gmail.com*

### **ABSTRACT**

In current research, QR Code system has become interesting. It has structural flexibility, so QR Code leads to so many diverse fields for research. It is also has fast readability. In spite of being higher data capacity, we need more data capacity than present. For increasing data capacity, there are so many methods and technologies are developed. Helping us in encoding the data in an efficient manner, it has become admired. The data capacity is limited because of its various data formats used and versions. In the paper, there are three steps in order to increase the data capacity. Input data can first be decomposed and then it is encoded into respective binary QR Codes. Finally, color QR code is created by multiplexing method. The propose of the paper is to use QR Codes as a message carrier for text. This paper suggests a technique for creating colored QR Code to increase the data capacity.

**KEYWORDS:** color QR code, data capacity, message carrier, multiplexing, QR code

### **1. INTRODUCTION**

Nowadays, although the use of digital messages is increased, paper works are still widely used in many places such as education and business and education. In business, people can put QR Image in the bag or pocket and print on the shirt or products, and take it everywhere without worrying about losing power or file corruption. QR Code, one of the most import parts of Automatic Identification and Data Capture (AIDC), 2D barcode, is a graphical image that store information both horizontally and vertically [1]. Japanese company Denso Wave Corporation developed it with the aim of improving upon the speed and data capacity of the traditional one dimensional bar codes.

To display text to the user, add a contact to the user's device, open a Uniform Resource Identifier (URI), or compose an e-mail or text message [2], QR codes are widely used. There are also some advantages in QR Code.

### **2. RELATED WORKS**

Max E. Vizcarra Melgar published the paper "CQR CODES: COLORED QUICK-RESPONSE CODES" in 2012 IEEE Second International Conference on Consumer Electronics. This paper proposes a new way to store/transmit information using a Colored QR Code structure. Instead of using only black and white modules,

The paper "Improved Color QR Codes for Real Time Applications with High Embedding Capacity" is published in International Journal of Computer Applications in April 2014 by M. Ramya, M. Jayasheela. In this paper the input element should be divided into 2 elements of each. The ASCII value of the first element should be added with the ASCII value of the next element and so on. All these values are grouped together to for the block of data. The colors such as cyan, yellow and magenta are assigned to all the bits and pixel values. By combining all the colors the final QR code is obtained in color.

### **3. LITERATURE REVIEW**

#### **3.1. Feature of QR Code**

In 1994, QR code (Quick Response Code) is developed by Denso Corporation. The smallest version of QR Code is 21×21 pixels, and the largest is 177×177. The size is called version. There are 40 versions in QR code, four levels of error correction. The maximum symbol size (the highest version) can encoding 7089 numeric data or 4296 alphanumeric

data. The highest level of error correction allows recovery of 30% of the symbol code words. [3]



Fig.1 QR Code Symbol

### 3.2. Advantages of QR Code

Advanced features of QR Code:

- 1) High capacity encoding of data
- 2) High-speed reading
- 3) Chinese encoding capability
- 4) Readable from any direction from 360°[4]

### 3.3. QR Code Symbol

Each QR code symbol consists of an encoding region and function patterns. Finder, separator, timing patterns and alignment patterns comprised function patterns.

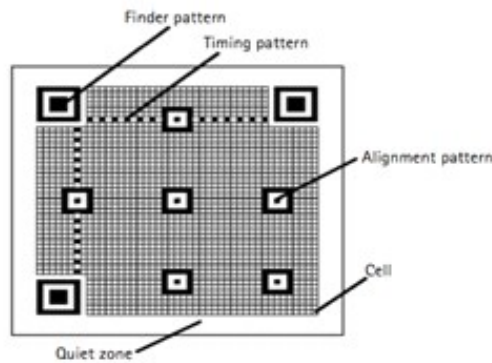


Fig.2 Structure of QR Code

Function patterns shall not be used for the encoding data. The finder patterns located at three corners of the symbol intended to assist in easy location of its position, size and inclination. [4]

## 4. PROPOSED SYSTEM

The new framework for color QR image construction enables extension of exiting binary QR images to color QR image. Data is encoded in three binary QR images which are then combined as the red (R), green (G), and blue (B). Color QR image construction can offer three times of data rates.

### 4.1. Proposed Framework for Encoding

Firstly, message has to be decomposed and put into the respectively sets. And these character sets are encoded into respective binary QR Codes. By using multiplexing method, we will get the color QR image. The proposed multiplexing method will be discussed in this sections. Finally, the color QR Code can be printed to send the receiver both physically and digitally. The printed color QR image can be readable. The proposed system is illustrated in the following:

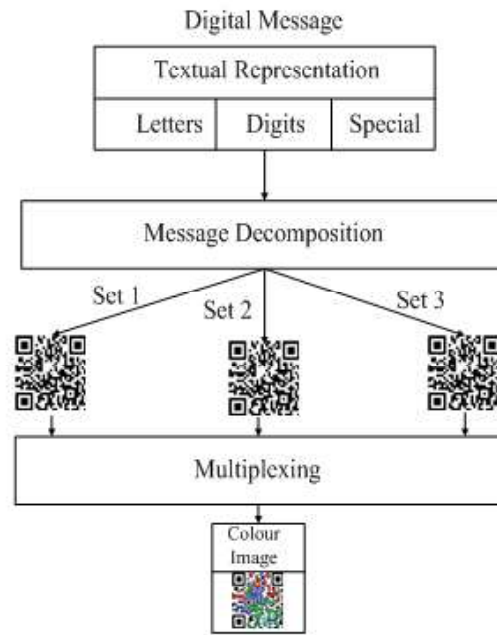


Fig.3 Proposed framework for encoding

## 4.2 Mathematical Expression for Multiplexing

A QR Code pattern is generated for each part in its standard form. Each pattern multiplexed and represented each module in QR Code with specific color.

The mathematical expression for multiplexing is shown in equation. There has a little ambiguous thing in this equation. Although  $VR_{(x,y)}$ ,  $VG_{(x,y)}$  and  $VB_{(x,y)}$  have 255 value, black color is set into the color QR image because a dark module is a binary one and a light module is a binary zero in QR standard. Having 0 value is vice versa.

$$VR_{(x,y)} = \begin{cases} \text{if } IMG1_{(x,y)} = 1, \text{ then } VR_{(x,y)} = 255. \\ \text{else } VR_{(x,y)} = 0. \end{cases}$$

$$VG_{(x,y)} = \begin{cases} \text{if } IMG2_{(x,y)} = 1, & \text{then } VG_{(x,y)} = 255. \\ \text{else} & VG_{(x,y)} = 0. \end{cases}$$

$$VB_{(x,y)} = \begin{cases} \text{if } IMG3_{(x,y)} = 1, & \text{then } VB_{(x,y)} = 255. \\ \text{else} & VB_{(x,y)} = 0. \end{cases}$$

```

if       $VR_{(x,y)} = VG_{(x,y)} = VB_{(x,y)} = 255,$ 
then    $Coloring_{(x,y)} = (0,0,0)$ 
elseif  $VR_{(x,y)} = VG_{(x,y)} = VB_{(x,y)} = 0,$ 
then    $Coloring_{(x,y)} = (255,255,255)$ 
else

```

$$Coloring_{(x,y,r)} = VR_{(x,y)}$$

$$Coloring_{(x,y,g)} = VG_{(x,y)}$$

$$Colorimg_{(x,y,b)} = VB_{(x,y)}$$

where  $x = 1, 2, 3, \dots, w$

$$y = 1, 2, 3, \dots, h$$

$w = \text{width\_of\_IMG}$


### *h = hight\_of\_IMG*

### 4.3. Proposed Color Expression

The QR Code is that code the data based on the position of black element. In the color QR code data based on the position of all color except white. Three binary QR Image are the same dimension and assume color channel(R, G, and B). If the respective positions of all binary QR images have the data, the position

of color QR image will be black element. If the position of first binary QR image is black and the others have no data, red color will represent on the position of color QR image. Color expression is shown in the following table.

Table 1. PROPOSED COLOR EXPRESSION

Channels			Color
Red	Green	Blue	
1	1	1	
1	0	0	
1	1	0	
1	0	1	
0	0	1	
0	1	1	
0	1	0	
0	0	0	

The process of changing color expression is shown in the following table.

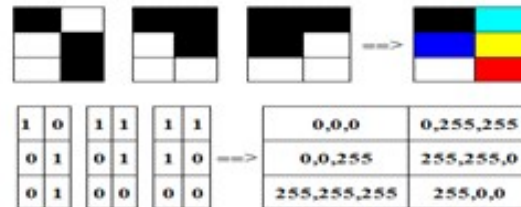


Fig.4 Proposed color expression

#### 4.4. Proposed Framework for Decoding

At the receiver side, color QR image is needed to insert to decode the colored QR image. Demultiplexing the color QR image, three binary QR images are given as color channel (Red, Green, Blue). The colored QR image is given by combining this three binary QR images which are represented color channels. The proposed system for decoding is illustrated in the following figure.

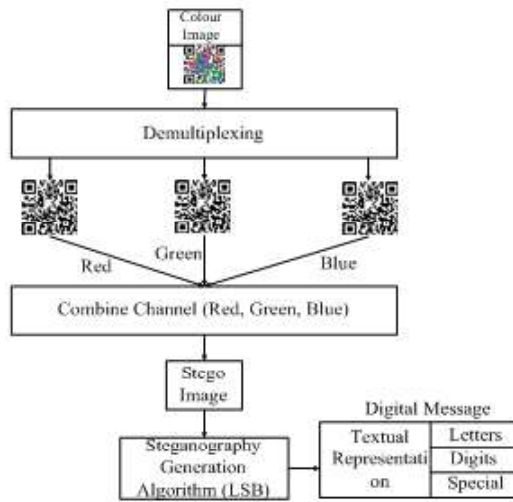


Fig.5 Proposed framework for decoding

#### 4.5 Mathematical Expression for Demultiplexing

The mathematical expression for demultiplexing is shown in following equation. This is a method which is one to decode QR color image to three binary QR images.

if  $Coloring_{(x,y)} = Coloring_{(x,y,g)} = Coloring_{(x,y,b)} = 0$ ,  
 then  $VR_{(x,y)} = VG_{(x,y)} = VB_{(x,y)} = 1$ .  
 elseif  $Coloring_{(x,y,r)} = Coloring_{(x,y,g)} = Coloring_{(x,y,b)} = 255$ ,  
 then  $VR_{(x,y)} = VG_{(x,y)} = VB_{(x,y)} = 0$ .

else

$VR_{(x,y)} = \begin{cases} \text{if } Coloring_{(x,y,r)} = 255, \text{ then } VR_{(x,y)} = 1. \\ \text{else } VR_{(x,y)} = 0. \end{cases}$

$VG_{(x,y)} = \begin{cases} \text{if } Coloring_{(x,y,g)} = 255, \text{ then } VG_{(x,y)} = 1. \\ \text{else } VG_{(x,y)} = 0. \end{cases}$

$VB_{(x,y)} = \begin{cases} \text{if } Coloring_{(x,y,b)} = 255, \text{ then } VB_{(x,y)} = 1. \\ \text{else } VB_{(x,y)} = 0. \end{cases}$

where  $x = 1, 2, 3, \dots, w$

$y = 1, 2, 3, \dots, h$

$w = \text{width\_of\_IMG}$

$h = \text{height\_of\_IMG}$

#### 5. Experimental Results and Discussion

The QR code has been implemented using C#.Net as shown in the flowing figure. The messages are embedded in above respective binary QR Codes. On the right color QR Code is created by using

multiplexing method. Demultiplexing it, three binary QR Codes will be got back.



Fig.6 Testing of changing three QR Code to a color QR Code

The following table shows the versions and maximum data capacity of QR Code. The smallest QR Codes are  $21 \times 21$  pixels, and the largest are  $177 \times 177$ . Although there are 40 version in QR code, some version and max bytes are shown in the table.

Table 2. LIST of DATA STORAGE of QR CODE

No	Version	Modules	Max Total Byte
1	1	21x21	17
2	5	37x37	106
3	10	57x57	271
4	15	77x77	520
5	20	97x97	858
6	25	117x117	1,273
7	30	137x137	1,732
8	35	157x157	2,303
9	40	177x177	2,953

The data capacity of color QR Codes are shown in the following table. The maximum bytes for smallest version is nearly 50 bytes and biggest is 8858 bytes.

Table 3. LIST of DATA STORAGE of COLOR QR CODE

No	Version	Modules	Max Total Byte
1	1	21x21	50
2	5	37x37	317

3	10	57x57	812
4	15	77x77	1559
5	20	97x97	2573
6	25	117x117	3818
7	30	137x137	5195
8	35	157x157	6908
9	40	177x177	8858

### 5.1 Comparison with Binary QR Code

The following figure shows the result of that color QR Code is compared with binary QR Code. Data capacity of color QR Code is nearly three time of binary QR Code.

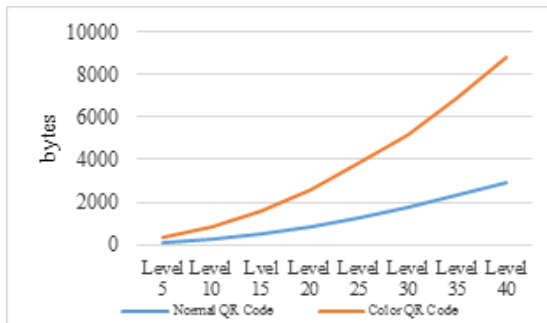


Fig.7 Comparison between binary QR Code and color QR Code in data capacity (Bytes)

The following table shows the testing result for data density in document, binary QR and color QR. Paper document (.docx) in general format with font Times New Roman, font size 12 and 1.5 line spacing would have a data density of average 52.3 bits per inch<sup>2</sup>. In the table we can found that comparison between these three types.

Table 4. LIST OF DATA DENSITY

No	Bytes	Bytes Per Area(inch <sup>2</sup> )		
		Document	Binary QR	Color QR
1	105	70	46.67	86.066
2	585	47.667	53.625	126.923
3	1732	51.548	54.81	149.827
4	2953	40.013	69.893	153.005

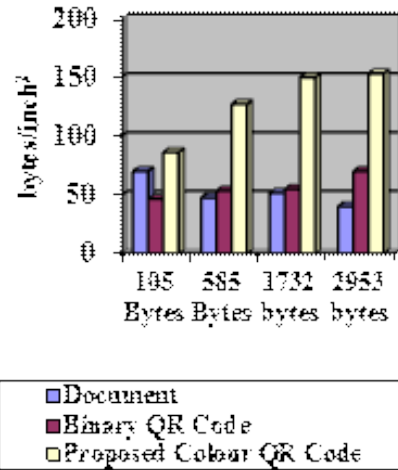


Fig.8 Comparison among document, binary QR Code and color QR Code in data density (bytes per inch<sup>2</sup>)

All experimental reports in this paper were run on Laptop, equipped with a 2.10 GHz Intel core 2 Duo processor, 2 GB RAM, running OS window 7. The application is implemented by using C#.NET 4.0. The following figure shows the variation of processing time by using proposed color QR Code.



Fig.11 Proposed color QR code

### 4.3. Discussion

The main propose of the paper research is to increase data capacity of QR Code. There are some technologies which were presented to solve this problem. Creating color QR code is proposed in order to increase data density on the same area of traditional QR Code. There are some challenges such as sufficient lighting required to capture QR Code.

In capturing and color extraction, other noises will also be got such as color missing due to light, charge-coupled device (CCD) of camera and printed paper.

## 5. CONCLUSIONS

The proposed framework in this paper helps us to encode data into color QR Code and combine three binary QR Codes to a color QR Code. The proposed system will help for not only messaging system but also other related occupation such as education, economic, transportation, etc. The data density of color QR Code is higher than paper document and traditional QR Code. Data capacity is also higher than binary QR Code. While it has some advantages, some noises still have such as light affect and color missing. These problems will be solved in the further.

## ACKNOWLEDGEMENT

We would like to express special thanks to all the people and all member of UCSMTLA organizing committee "Journal of Research & Applications (JRA), UCSMTLA, 2019, Volume-01, Issue-01" for paper invitation. We are grateful to all teachers and reviewers who provided helpful during the development of this paper and the faith they have had in me

## REFERENCES

- [1] J.-A. Lin and C.-S. Fuh, "2D Barcode Image Decoding," *Mathematical Problems in Engineering*, 2013. [Online]. Available: <https://www.hindawi.com/journals/mpe/2013/848276/>. [Accessed: 13-Dec-2019].
- [2] "QR Code Recognition Based on Image Processing | Qr Code | Barcode," *Scribd*. [Online]. Available: <https://www.scribd.com/document/239468382/QR-Code-Recognition-Based-on-Image-Processing>. [Accessed: 13-Dec-2019].
- [3] "QR Code Tutorial - Thonky.com." [Online]. Available: <https://www.thonky.com/qr-code-tutorial/>. [Accessed: 13-Dec-2019].
- [4] Y. Liu and M. Liu, "Automatic Recognition Algorithm of Quick Response Code Based on Embedded System," in *Sixth International Conference on Intelligent Systems Design and Applications*, 2006, vol. 2, pp. 783–788.
- [5] M. Querini and G. Italiano, "Reliability and Data Density in High Capacity Color Barcodes," *Comput. Sci. Inf. Syst.*, vol. 11, pp. 1595–1615, Oct. 2014.
- [6] A. Singh, "A REVIEW: QR CODES AND ITS IMAGE PRE-PROCESSING METHOD," vol. 5, no. 6, p. 6.
- [7] S. Kong, "QR Code Image Correction based on Corner Detection and Convex Hull Algorithm," *J. Multimed.*, vol. 8, no. 6, pp. 662–668, Nov. 2013.
- [8] J. Zhou, Y. Liu, and A. Kumar, "Research on Distortion Correction of QR Code Images," vol. 3, no. 1, p. 6, 2012.
- [9] N. Victor, "Enhancing the Data Capacity of QR Codes by Compressing the Data before Generation," *Int. J. Comput. Appl.*, vol. 60, pp. 17–21, Dec. 2012.
- [10] S. Chen, "Evaluation of Two-Dimensional Codes for Digital Information Security in Physical Documents," p. 58.

## **SPAM DETECTION USING HYBRID METHOD WITH NLP AND MACHINE LEARNING TECHNIQUE**

**Khin Myat Nwe Win <sup>(1)</sup>, Aye Mya Sandar <sup>(2)</sup>, Yin Myo Kay Khaing Thaw <sup>(3)</sup>**

<sup>(1)</sup> Faculty of Computer Science, University of Computer Studies (Mandalay), Myanmar

<sup>(2)(3)</sup> University of Computer Studies (Mandalay), Myanmar

<sup>(1)</sup>khinmyatnwewin@gmail.com

### **ABSTRACT**

Nowadays, Internet is everything for everyone because most of the people surf the Internet for various kinds of reasons. Among different Internet activities such as commenting, posting, reacting and some advertisements and writing spam post comments on some posts of social networks such as Facebook, Instagram or videos such as YouTube, Netflix, etc. In order to get rid of unnecessary and un-useful comments from social networks, in this paper, we present a anti-spam detection algorithm for human users' YouTube comments. This paper uses a new hybrid algorithm that mixes the classification abilities of logistic regression and feature selection assisted relatedness relations process. For the experiment works, we use a popular spam database for YouTube online videos. The results are compared with classification benchmarks metrics such as accuracy, precision and recall to compared with other contemporary anti-spam approaches in terms of classification alone and both classification and NLP based semantic relatedness exploration based on the features. According to the results, our results outperform the others.

**KEYWORDS:** *anti-spam classification, logistic regression, NLP-based semantic relatedness, features selection*

### **1. INTRODUCTION**

Spam filtering has gained much attention in text mining and classification area since many years ago. Text classification, also known as text categorization, has become very popular since the evolution of the Internet. The aim of text classification is to assign

electronic documents into pre-defined set of categories. It has various application areas such as sentiment classification [1], medical document classification [2], news classification [3], spam e-mail filtering [4], spam short message filtering [5], and spam comment filtering on social media [6]. However, it migrates to Web related things' classification matters has reached to popularity only in very recent years. Due to the rise in the usage of social media platforms such as YouTube, Facebook, and Twitter, the number of different social media. While there are many studies on classification of spam e-mails and short text messages, comment anti-spam detection and classification on YouTube is relatively a new topic due to its limited numbers of annotated datasets.

The differences between traditional text classification and YouTube comment classification is relatively different because the former one works on the textual information, which are most of formal languages such as papers, magazines and online stores, etc. Conversely, the latter one has to deal with most volatile languages because online users are more likely to use Slang, idioms and emotion icons, etc., which is relatively complicated depending on the condition they want to say. Therefore, to the best of our knowledge, solely solving this problem with machine learning techniques cannot reach the acceptable results because human languages are so sparse and changes depending on the users' tastes from time to time.

To address the abovementioned challenges in this YouTube comment spam detection area, we propose a new hybrid anti-spam detection algorithm that takes the advantages of heuristics of machine

learning algorithms and also NLP based language analysis tasks. In statistical data, linear regression is a linear approach to explore the relationship between a scalar response and one or more explanatory variables. Therefore, among many classification algorithms linear regression approach is chosen due to high suitability with our current YouTube comment anti-spam detection system.

To the best of our knowledge, we are the first of bringing them together in this YouTube spam detection problem. The paper proposes a hybrid anti-spam with a following main steps:

- Logical regression algorithm that works on numeric number
- NLP techniques are used to covert the occurrence of the words and their cooccurrences of word using bag of words and word embeddings
- We then use tf-idfs and also their relations from language points of view are analysis depending on the features of the user's comment posts.

The remainder of the paper is presented with the following format. Related work is described in Section 2 while the proposed system is widely used in Section 3. The experimental works are performed in Section 4 and the paper is concluded in Section 5.

## **2. RELATED WORK**

Many studies on the detection of malicious items on the social network have been conducted and this includes mining the media content and analyzing it (i.e. Content-based) [1] [2]. For instance, mining the comments provided by users and learn the pattern to detect malicious contents. However, mining the keywords requires large computational cost and it is limited to the listed words only. As Comments could be written using formal and informal language, relying on keywords would not be efficient.

There are machine learning approaches that have explored for this spam detection and classification problems. For example, rule-based methods, such as Ripper [3], Decision tree, and Rough Sets, have been used in [4], [5] and [6]. Nonetheless, pure rule-based methods was unable to achieve high performance because spam emails cannot easily be covered by rules, and rules don't provide any sense of degree of evidence.

Statistical or computation-based methods have proven more successful, and are generally adopted in mainstream work. Bayesian classifiers are the most

widely used method in this field. Sahami et al. [7] used Naïve Bayes with an unpublished email test collection. In their work some non-textual features (e.g., the percentage of non-alphanumeric characters in the subject of an email) were found to improve the final performance.

Video spammers are motivated to perform spamming in order to promote specific content [8]. A video spam occurs when a video posted as a response to an opening video. Whereas, the content is completely unrelated to the video's title [9]. Since users cannot easily identify a video spam before watching at least a segment of it, users will waste their system resources, in particular, the bandwidth. Furthermore, it compromises user patience and satisfaction with the system. Thus, identifying video spam is a challenging problem in social video sharing systems.

YouTube also faces malicious users that publish low quality content videos, which it is known as video spam. There are some studies in literature to find efficient ways to handle this activity through classification methods and feature extraction from metadata, such as title, description and popularity numbers [10].

As noted by the study [11], spam filtering task slightly differs from similar text categorization problems. They claim undesired messages have chronological order and their characteristics may change according to that. It also explains that cross-validation is not recommended, because earlier samples should be used to train the methods, while newer ones should be used to test them. Furthermore, in spam filtering, errors associated with each class should be considered differently, because a blocked legitimate message is worse than an unblocked spam.

## **3. HYBRD SPAM DETECTION SYSTEM**

Unsolicited pile of comments in online video such as YouTube, also known as spam comments, are a regular occurrence for every video uploader and video viewers. Spam filtering is a process of distinguishing between spam comments and regular comments. The goal of spam filtering is to determine whether a comment is spam or not spam, useful or un-useful to the video viewers by filtering out the spam comment, resulting in a spam-free in-box for the user.

In this paper, we introduce a hybrid spam detection system with machine learning algorithm

called logistic regression and NLP-based language handling methods.

### **3.1 NLP-based Language Model for Feature Selection**

In analyzing the human's language, which is kind of sparse, it needs a thorough analysis from language point of view besides logistics regression. In this paper, the idea of considering bag of words and word embedding is brought in this spam detection system in order to explore more related or unrelated language pair using NLP tools and techniques.

The main purpose of using NLP tools is to extract the useful features from the raw users' video comments, that can come in various topics, contents and different word combinations. To extract the features, we first find the bag of words.

#### **3.1.1 Bag of Words**

According to the-state-of-art of the NLP field, Embedding is the success way to resolve text related problem and outperform Bag of Words (BoW). Indeed, BoW introduced limitations such as large feature dimension, sparse representation etc. Among different BoW methods namely count occurrence, normalized count occurrence, and TF-IDF occurrences, this paper considers TF-IDF occurrences because it is most simple and popular content filtering approach in text classification. The word occurred in comments of all users are analyzed using Tf-Idf calculation in eq (1).

$$w_{(x,y)} = \frac{tf_{(x,y)} \times \log(N/df_{(x,y)})}{(1)}$$

#### **3.1.2 Word Embedding**

Word embedding is one of the most popular representation of document vocabulary. It is capable of capturing context of a word in a document, semantic and syntactic similarity, relation with other words, etc.

Among different word embedding, we utilize Word2Vec word embedding, which is a very popular approach in analyzing the relatedness and semantic meaning of the pairs of words in NLP.

We would have a vector of zeros except for the element at the index representing the corresponding word in the vocabulary. That particular element would be one. The encodings below would explain

this better. It transforms a vector that represents the word contents found in all documents considered in this application.

This objective is to have words with similar context occupy close spatial positions. Mathematically, the cosine of the angle between such vectors should be close to 1, i.e. angle close to 0.

Word2vec is a two-layer neural net that processes text. Its input is a text corpus and its output is a set of vectors: feature vectors for words in that corpus. While Word2vec is not a deep neural network, it turns text into a numerical form that deep nets can understand.

The purpose and usefulness of Word2vec is to group the vectors of similar words together in vector space. That is, it detects similarities mathematically. Word2vec creates vectors that are distributed numerical representations of word features, features such as the context of individual words. It does so without human intervention.

### **3.2 Logistic Regression**

In statistics, the logistic model (or logit model) is used to model the probability of a certain class or event existing such as win/lose, pass/fail or healthy/sick. Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression).

Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail which is represented by an indicator variable, where the two values are labeled "0" and "1". In the logistic model, the log-odds for the value labeled "1" is a linear combination of one or more independent variables, which is also known as predictors; the independent variables can each be a binary variable (two classes, coded by an indicator variable) or a continuous variable.

In logistics regression, there are two main steps in calculating the probabilities of elements of classifying into classes (1 (spam) or 0 (ham or non-spam)).

**Step 1.** Classifying inputs to be in class zero or one.

It needs to compute the probability that an observation belongs to class 1(spam in this paper) using the Logistic Response Function. In this case, our z parameter is, as seen in the below given logit function.

$$P(y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \dots + \beta_n x_n)}} \quad (2)$$

The above equations can be understood in terms of logarithmic odds, which is a kind of understanding the probabilities to classify elements into classes (1 or 0) is by using Odds shown in eq (3).

$$\text{odds} = \frac{P(y=1)}{P(y=0)} = \frac{\text{The Odds will be } >1 \text{ when there is a higher probability of predicting } y=1}{\text{The Odds will be } <1 \text{ when there is a higher probability of predicting } y=0} \quad (3)$$

These Odds which resembles similarity to a linear regression is called the logit. So, a logit is a log of odds and odds are a function of P in eq (4). In logistic regression, we find the log value as follows:

$$\text{logit}(P) = a + bX \quad (4)$$

#### Step 2: Defining a boundary values for the odds

A threshold value can be chosen as per the business problem we are trying to solve, generally which is circled around 0.7. So, if your probability values come out to be  $>0.7$  we can classify such observation into class 1 type (spam), and the rest into class 0(non-spam).

In this paper, we give alternative threshold value to measure its performance and record the average value for all test and chose the best threshold value. According to the result, the best threshold is 0.7.

### 3.3 System Overview

The system overview is illustrated in Figure 1 as shown in below. There are four main components in the system namely, pre-processing, NLP based language modelling, machine learning techniques and data fusion methods.

The system starts spam detection process after accepting the user input, which is YouTube's comment in this paper. The user input can contain various kinds of words, that is not suitable for the system's detection process. We call them as noises and pre-processing step removes them by removing stop-words, URL links, emotion icons, etc.

Afterwards, the system uses NLP-tools to extract the most suitable features in detecting the comments as spam or not. As the first step of NLP processes, a bag of words is collected from the input data and also dataset data so that the occurrences of the words are calculated using Tf-idf. It then considers word embedding using a popular technique called Word2Vec to understand the relations between the words and their semantic meaning. Depending on that, we extract only most suitable features depending on their semantic meanings and Tf-idf occurrences. We give the more weight if they have more relatedness and more occurrences.

The third part of the system is to utilize machine learning techniques in order to find the probability values of the words depending on the inputs and dataset. The values obtained from NLP and Machine learning processes are then combined as data fusion steps. In this case, the values of each process are weighted with  $w_1$  and  $w_2$  for each value  $X_1$  and  $X_2$  for the final classification value  $C_v$ . The final value will be in the range of 0 to 1 and the less value means non-spam while the more value is spam depending on its threshold value 0.7. The system finally outputs the input comment as spam or non-spam.

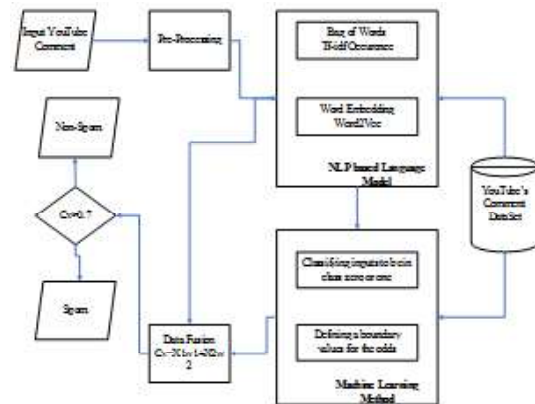


Figure 1. System Architecture

#### 4. EXPERIMENTAL WORKS

##### 4.1 DataSet Setting

In order to perform the experiments for our proposed system, we first set up the dataset setting and development setting.

**Table 1.** DatSet Packages

Data Set	YouTubeID	Spam	Ham	Total
PSY	9bZkp7q19f0	175	175	350
Katy Perry	CevxZvSjLk8	175	175	350
LMF AP	KQ6zr6kCPj8	236	202	438
Eminem	uelHwf8o7_U	245	203	448
Shakira	pRpeEdMmmQ0	174	196	370

First of all, a popular YouTube comment data is downloaded from UCI machine learning repository [12], which crawls the comments from celebrity YouTube account, which are PSY, KatyPerry, LMFAP, Eminem, Shakira. In each account, two types called Spam and Ham are prepared and each has around 350-450 comments in total.

For the development setting, we experiment this proposed system with Python Programming and uses NLP library supported by NLP such as pytorch, sklearn, Sciki-learn, etc.

##### 4.2 Experimental Results

In analyzing the classification problem, spam or non-spam in this paper, the traditional analysis process evaluates confusion matrix as Table 2.

The **confusion matrix** is another metric that is often used to measure the performance of a classification algorithm.

· **True positives (TP):** the cases for which the classifier predicted 'spam' and the YouTube comments were actually spam.

· **True negatives (TN):** the cases for which the classifier predicted 'not spam' and the YouTube comments were actually real.

· **False positives (FP):** the cases for which the classifier predicted 'spam' but the YouTube comments were actually real.

· **False negatives (FN):** the cases for which the classifier predicted 'not spam' but the YouTube comments were actually spam.

**Table 2.** Confusion matrix

	Predicted Non-Spam Comment	Predicted Spam Comment
Actual Non-Spam Comment	True Negatives (TN)	False Positives (FP)
Actual Spam Comment	False Negatives (FN)	True Positives (TP)

Depending on confusion results, we then evaluate accuracy (4), precision (5) and recall (6) results as listed in following table. In evaluating their results, we measure our own system with different parameters such as only NLP based methods, only machine learning methods and mixing up of them so that the effectiveness of our system can be revealed.

**Table 3.** Experimental Results Comparison

	Spam Detection with only NLP techniques	Spam Detection with only Machine learning	Spam Detection with both NLP and Machine
Accuracy	76%	82%	96%
precision	73%	83%	95%
recall	67%	77%	89%

As shown in Table 3, our system with complete consideration of both NLP and Machine learning techniques get much better results than the others, that considers only one approach. After comparing ourselves under different parameters,

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

In order to compare with other contemporary works, we try to compare our complete algorithm with other study [1]. The results are listed in Table 4

and as can be seen, our system outperforms the other approach due to full consideration of language modelling and classification techniques.

**Table 4.** Evaluation results with other work

	Our YouTube Spam Detection	YouTube Spam Detection [1]
Accuracy	82%	72%
precision	83%	78%
recall	77%	63%

## 5. CONCLUSIONS

Nowadays, social media networks have become extremely popular and this creates the opportunity for the malicious user to publish unwanted content such as video spam. In this paper, we propose a hybrid spam detection system by combining machine learning algorithm and NLP-based language understanding model so that the performance of spam detection can be increased. As the future work, we plan to extend this spam detection system to be more robust with all possible data such as dirty data which occurs missing values and also less resource area.

## REFERENCES

- [1] T. Alberto, J. Lochter, and T. Almeida, "TubeSpam: Comment Spam Filtering on YouTube," 2015, pp. 138–143.
- [2] Y. Yusof and O. Hadeb, "DETECTING VIDEO SPAMMERS IN YOUTUBE SOCIAL MEDIA," Apr. 2017.
- [3] W. W. Cohen, "Fast Effective Rule Induction," p. 10.
- [4] H. Drucker, D. Wu, and V. N. Vapnik, "Support Vector Machines for Spam Categorization," *Neural Netw. IEEE Trans. On*, vol. 10, pp. 1048–1054, Oct. 1999.
- [5] J. Gomez Hidalgo, "Evaluating cost-sensitive Unsolicited Bulk Email categorization," 2002, pp. 615–620.
- [6] B. Wang, G. J. F. Jones, and W. Pan, "Using online linear classifiers to filter spam emails," *Pattern Anal. Appl.*, vol. 9, no. 4, pp. 339–351, Oct. 2006.
- [7] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, "A Bayesian Approach to Filtering Junk E-Mail," p. 8.
- [6] V. Umayaparvathi and K. Iyakutti, "Automated Feature Selection and Churn Prediction using Deep Learning Models," vol. 04, no. 03, p. 9.
- [7] T. Vafeiadis, K. I. Diamantaras, G. Sarigiannidis, and K. Ch. Chatzisavvas, "A comparison of machine learning techniques for customer churn prediction," *Simul. Model. Pract. Theory*, vol. 55, pp. 1–9, Jun. 2015.

## MEAN SQUARE ERROR EFFECTS OF DATA HIDING WITH EDGE DETECTION ALGORITHMS

Zin Mar Htun<sup>(1)</sup>, Aye Zarchi Minn<sup>(2)</sup>, Aung Kyaw Oo<sup>(3)</sup>

<sup>(1)(2)(3)</sup>Department of Information Technology, Technological University (Taunggyi)

Taunggyi, Myanmar

<sup>(1)</sup>zinmarhtun.myanmar@gmail.com, <sup>(2)</sup>ayemin.rose@gmail.com, <sup>(3)</sup>aung15kyaw.ak@gmail.com

### ABSTRACT

With the development of communication over computer network, the safety of information has grown into a main issue. Steganography or data hiding is the art and science of invisible communication and the attacker cannot know the presence of the message that must be kept secret. Data hiding using edge detection techniques is a popular research today because it provides more secure than that without edge detection. However, the researchers need to investigate for affecting other factors such as Mean Square Error (MSE). Mean Square Error is the important issues for data hiding because the less MSE error the system gets, the more secure it provides in hiding. Therefore, the research compares the effects of edge detection methods on the image steganography that makes use of Least Significant Bit (LSB) algorithm for hiding the data into an image by varying the size and type of images.

**KEYWORDS:** *Least Significant Bit, Data Hiding, Edge Detection Algorithm, Mean Square Error*

### 1. INTRODUCTION

Security is getting compromised more often during an exchange process between sender and receiver. Data hiding called steganography is the practice of concealing a file, message, image or video within another file, message, image, or video. Computer vision image processing can be done through digital images which include gray scale images with edges. Edges can be detected in many various ways: Robert, canny, rewtitt, laplace, sobel edge detection techniques. The edges of an image can be used very efficiently for hiding data with many edge detection methods. LSB is one of the ancient

steganography algorithms that embed the messages bits into the stegno-image. The extraction phase is the opposite of the embedding phase.

Data hiding using edge detection provides more security than that without edge detection. However, it has less embedding capacity. Therefore the researcher needs to investigate other factors such as MSE, PSNR.

Aim of the paper is to compare the effects of edge detection methods on the image steganography. Objectives are

- To get good capacity for carrying the amount of secret message
- To make more security and robustness along the network communication
- To learn edge detection techniques
- To know how to hide and extract data using LSB techniques
- To make hybrid method using image and data hiding techniques for more secure

[1] presented comparison of various Edge Detection Techniques for maximum data hiding using LSB algorithm. [2] described Sobel edge detection technique implementation for image steganography analysis.

[3] described that LSB method is used for embedding and extracting. Sobel and Canny Edge detection algorithms are used to find the positions for hiding data.

Consequently, steganography is used to embed secret information in the least significant bit of every pixel. However, this technique [4] cannot resist statistical attacks. A few years later, some works of research [5] [6] [7] proposed that under the condition which isn't detectable, more information can be embedded in sharp areas than in smooth areas.

An Image Steganography based on a Parameterized Canny Edge Detection Algorithm [8]. It was intended to hide top-secret data into pixels of the image that were chosen the boundaries of objects detected in the image. It gave high imperceptibility but low capacity.

## 2. METHODOLOGY

There are two main methodology; steganography and Edge Detection Techniques. The main methodology for steganography is LSB. There are two methodologies in edge detection techniques.

### 2.1. Edge Detection Techniques

Edge detection is a problem of fundamental importance in image analysis. In typical images, edges characterize object boundaries and useful for segmentation, registration, and identification of objects in a scene. Edge detection of an image reduces significantly the amount of data and filters out information that may be regarded as less relevant preserving the important structural properties of an image. A theory of edge detection is presented. The analysis proceeds in two parts:

- Intensity changes, which occurs in a natural image over a wide range of scales are detected separately at different scales. An appropriate filter for this purpose at a given scale is found to be the second derivative of a Gaussian. Intensity changes at a given scale are best detected by finding the zero values of image. The intensity changes discovered in each of the channels are represented by oriented primitives called zero-crossing segments.

- Intensity changes in images arise from surface discontinuities or from reflectance or illumination. Boundaries and these all have the property that the boundaries are spatially localized. Because of this, the zero crossing segments from the different channels are not independent and rules are deduced for combining them into a description of the image. This description is called the raw primal sketch.

Edge detection aims at identifying points in a digital image at which the image brightness changes

sharply or more formally has discontinuities. These edge detectors are:

- Sobel edge detector
- Prewitt edge detector
- Canny edge detector is non-maximal suppression of local gradient magnitude.
- Zero Crossing Detectors is used the laplacian of Gaussian operator.

In Sobel Edge Detection, the operator consists of a pair of  $3 \times 3$  convolution kernels as shown in Figure 1. One kernel is simply the other rotated by  $90^\circ$ . These kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation (call these  $G_x$  and  $G_y$ ). In Fig. 1, masks for Sobel operator are shown.

1	0	-1
2	0	-2
1	0	-1

(a)  $G_x$

+1	+2	+1
0	0	0
-1	-2	-1

(a)  $G_y$

Fig.1. Masks used by Sobel Operator

-1	0	+1
-2	0	+2
-1	0	+1

(a)  $G_x$

-1	-2	-1
0	0	0
1	2	1

(a)  $G_y$

Fig.2. Masks used by Canny Operator

The Canny edge detector is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images. The steps in the Canny edge detector are as follows:

- Smooth the image with a two dimensional Gaussian. In most cases the computation of a two dimensional Gaussian is costly, so it is approximated by two one dimensional Gaussians, one in the x direction and the other in the y direction.
- Take the gradient of the image. This shows changes in intensity, which indicates the presence of edges. This actually gives two results, the gradient in the x direction and the gradient in the y direction.
- Non-maximal suppression. Edges will occur points the where the gradient is at a maximum. Therefore, all points not at a maximum should be suppressed. In order to do this, the magnitude and direction of the gradient is computed at each pixel. Then for each pixel check if the magnitude of the gradient is greater at one pixel's distance away in either the positive or the negative direction perpendicular to the gradient. If the pixel is not greater than both, suppress it.

## 2.2. Least Significant Bit (LSB)

Least Significant Bits substitution is one of the most public hiding method due to its easiness. The secret message is considered as bit stream and will be concealed into cover image by varying the LSBs of the cover image with the boot stream of the secret file. With only 1 bit substitution, the change between the cover and stego image is scarcely obvious by human visual system. However, when the extent of secret message is great, the hiding capacity will increase. Therefore, more bits will be used to cover the large secret message and consequently, more degradation may announce to the stego image and hence effects its imperceptibility.

## 2.3. Mean Square Error

The mean-square error (MSE) is used to compare image compression quality. The MSE represents the cumulative squared error between the compressed and the original image. The lower the value of MSE, the lower the error.

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad \text{Eq 1}$$

## 3. Test and Results

The proposed system flowchart is as shown in Fig 3. The proposed system is divided by two

functions; embedding and extracting. In embedding, original message is converted to grayscale image and find edges by using edge detection techniques. And then message is hidden in edges. Finally, the stegano image is gotten.

In extracting, stegano image is converted to grayscale images and finds edges using the same edge detection in embedding. And then the system extracts the message form edges.

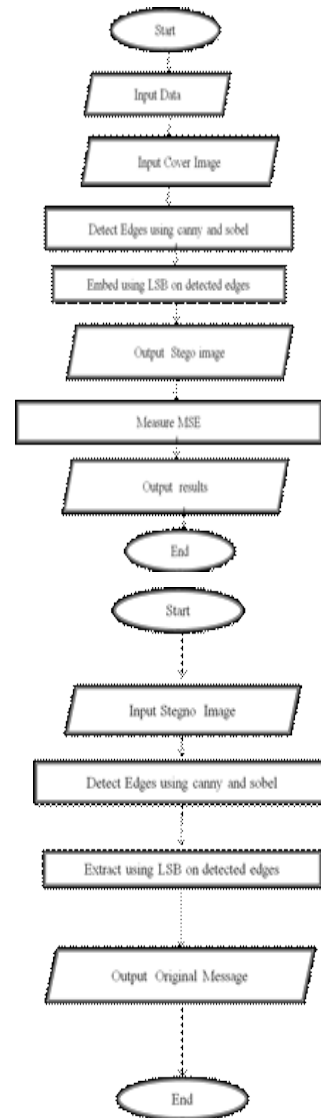


Fig.3 Proposed System Flowchart (a) for Embedding (b) for Extracting



(a) Image 1 (512'512)



(b) Images 2 (1024x1024)

Fig 4. Comparison of Original and Stego images

There are hiding 3 message sizes such as 100 byte, 200bytes and 300 bytes in 3 JPEG images such as 128x128 image, 512x512 image and 1024x1024 image by varying the size of images.

Some of the tested original images and stego images are shown in Fig. 4.

Table 1MSE Results for original LSB

		MSE using only LSB
Message 1	128x128 image	0.018
Message 1	512x512 image	0.019
Message 1	1024x1024 image	0.023
Message 2	128x128 image	0.041
Message 2	512x512 image	0.043

Message 2	1024x1024 image	0.0415
Message 3	128x128 image	0.0612
Message 3	512x512 image	0.0665
Message 3	1024x1024 image	0.0645

Table 2 MSE Results for Combining LSB and Canny

		MSE using LSB and Canny
Message 1	128x128 image	0.021
Message 1	512x512 image	0.018
Message 1	1024x1024 image	0.021
Message 2	128x128 image	0.039
Message 2	512x512 image	0.0425
Message 2	1024x1024 image	0.0431
Message 3	128x128 image	0.0623
Message 3	512x512 image	0.0612
Message 3	1024x1024 image	0.0624

Table 3 MSE Results for Combining LSB and Sobel

		MSE using LSB and Sobel
Message 1	128x128 image	0.017
Message 1	512x512 image	0.019

Message 1	1024×1024 image	0.022
Message 2	128×128 image	0.038
Message 2	512×512 image	0.0421
Message 2	1024×1024 image	0.044
Message 3	128×128 image	0.064
Message 3	512×512 image	0.0657
Message 3	1024×1024 image	0.0631

As shown in Fig 4, although the sizes of images vary, the level of presence of hiding data is not affected. Other testings are the same. However the statistics calculations gives the errors values between original and stego images.

Results of the MSE values when hiding 3 messages in various size of images as shown in Table 1, Table 2 and Table 3.

#### 4. ANALYSIS

According to the results shown in Table 1, MSE of the original LSB, LSB using Canny and LSB using Sobel are not much difference. The more the size of increases, the higher the MSE will be large.

Although varying the size of the images, MSE is not difference because all methods are use the same number of pixels but the different positions of the pixels.

#### 5. CONCLUSIONS

Finally the author conclude that data hiding combining edge detection methods gives more security because these methods do not use serial pixel for hiding data. These methods hide the message the positions of the edges. And so the attacker do not steal the message without knowing the edge positions. Moreover the paper recommend that combining LSB and edge detection techniques should use in data hiding for security and do not affect in MSE.

#### ACKNOWLEDGEMENT

The author would like to thank people to help for writing this paper.

#### REFERENCES

- [1] S. Sarkar and A. Basu, "Comparison of various Edge Detection Techniques for maximum data hiding using LSB Algorithm," vol. 5, p. 6, 2014.
- [2] S. GL and B. E, "Sobel edge detection technique implementation for image steganography analysis," *Biomed. Res.*, Jan. 2018.
- [3] S. Kaur and I. Singh, "Comparison between Edge Detection Techniques," *Int. J. Comput. Appl.*, vol. 145, no. 15, pp. 15–18, Jul. 2016.
- [4] B.Chitradevi, N.Thinaharan, and M.Vasanthi, "Data Hiding Using Least Significant Bit Steganography in Digital Images," Jan. 2017.
- [5] N. Jain, S. Meshram, and S. Dubey, "Image Steganography Using LSB and Edge – Detection Technique," vol. 2, no. 3, p. 6, 2012.
- [6] W.-J. Chen, C.-C. Chang, and T. H. N. Le, "High payload steganography mechanism using hybrid edge detector," *Expert Syst. Appl.*, vol. 37, no. 4, pp. 3292–3301, Apr. 2010.
- [7] A. Ioannidou, S. T. Halkidis, and G. Stephanides, "A novel technique for image steganography based on a high payload method and edge detection," *Expert Syst. Appl.*, vol. 39, no. 14, pp. 11517–11524, Oct. 2012.
- [8] C.-C. Chang and H.-W. Tseng, "A steganographic method for digital images using side match," *Pattern Recognit. Lett.*, vol. 25, no. 12, pp. 1431–1437, Sep. 2004.
- [9] D.-C. Wu and W.-H. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern Recognit. Lett.*, vol. 24, no. 9, pp. 1613–1626, Jun. 2003.
- [10] C.-M. Wang, N.-I. Wu, C.-S. Tsai, and M.-S. Hwang, "A high quality steganographic method with pixel-value differencing and modulus function," *J. Syst. Softw.*, vol. 81, no. 1, pp. 150–158, Jan. 2008.

- [11] C.-C. Thien and J.-C. Lin, "A simple and high-hiding capacity method for hiding digit-by-digit data in images based on modulus function," *Pattern Recognit.*, vol. 36, no. 12, pp. 2875–2881, Dec. 2003.
- [12] Y. Bassil, "Image Steganography based on a Parameterized Canny Edge Detection Algorithm," *Int. J. Comput. Appl.*, vol. 60, no. 4, pp. 35–40, Dec. 2012.
- [13] R. Wazirali and Z. Chaczko, "Hyper edge detection with clustering for data hiding," Jan. 2016.