A Review of Image Steganography Tools

Anal Kumar[*] and Hermann Jamnadas[*]
Fiji National University, Nadi, Fiji

Vishal Sharma[†]
Fiji National University, Nasinu, Fiji

S M Muyeen[‡],
Qatar University, Doha 2713, Qatar

A. B. M Shawkat Ali[§]
University of Fiji, Lautoka, Fiji

Abstract

Steganography is viewed like cryptography in one sense and both of them aim to guarantee the privacy of data. With the increasing application of steganography in the digital world, many issues must be understood in computer forensic inspection. There is a wide variety of tools and techniques, with their own focus and weaknesses. Stable changes must be made and more recent adaptations have been made. Initially, Steganography overview provided computer forensics experts in this field with knowledge and understanding of steganography. Consequently, the strength of most steganography tools depends on how well the software shrouds data or hides identification. As such it is basic to audit and comprehend which image-based steganography tool might be reasonably best and what are a portion of the basic pointers to recognizing concealed information in records. This research has been carried out through downloading and testing different image based steganography tools and assessing many articles and conference procedures. Various image steganography tools that were randomly selected and the list is not intended to be an exhaustive one consisting of all possible image steganography tools. Additionally, there is a filter that was applied to separate all steganography tools that were designed in Java and were windows based with GUI interface. These filtered Java and Windows based steganography tools were tested to see the encode and decode time difference, the visual differences and file size variance. It tends to be plainly seen that many picture steganography tools have blemishes that take into consideration the simple recognition or sign of a concealed document in a stego-picture. As such it is basic that individuals engaged with data security use those picture steganography tools that will diminish the opportunity of the message being identified since the strength of steganography is not to make the message ambiguous but instead to cause it to appear as though it was never there in any case. The authors posit that this research will be valuable for those keen on utilizing or testing image-based steganography tools.

Key Words: Image steganography, stego-picture, steganalysis, steganography tools

1 Introduction

Security of information has been a core concern for most of humanity. Many relied on elements of cryptography to encrypt secret messages and communiques to prevent malicious parties from taking advantage of successful interceptions of such messages [19]. However, cryptography does not hide the communication from other parties making it possible for others to attempt to alter the contents of the encrypted message [7]. Hence, steganography is useful as it allows for such communication to be hidden from others by embedding a secret message inside a multimedia file such as an image file, audio file, or a video file [20]. Images are a suitable medium to act as "carriers" since they are one of the most prevalent types of media and can store substantial amount of data without affecting image quality [1]. Image steganography tools takes a secret message or file and embeds it in a cover image to create a stego-imag [1, 18]. The strength of the image steganography lies in the fact that nobody is aware or is suspicious of the existence of hidden data in a nondescript image file. Hence a steganography system or tool will become useless if the "carriers" are suspected of harboring a secret message [4, 18]. It is essential therefore to use only those image steganography tools that will yield the least amount of suspicion.

This paper will thus begin with a brief overview of the steganography tools to be reviewed, the tests to perform on

[*] Department of Computer Science and Information Systems. Email: anal.kumar@fnu.ac.fj

[†] Department of Computer Science and Information Systems.

[‡] Department of Electrical Engineering.

[§] Department of Computer Science and Mathematics.

such tools and the analysis of the results that will help determine which of the image steganography tools will yield stego-images that are nondescript in nature.

2 Image Steganography Tools Overview

2.1 Hide 'N' Send

Hide'N'Send is a clever little application that can be utilized to cover text records containing private messages, data and passwords by concealing them inside JPEG pictures, which you can share through email or some other medium. It's additionally important that the application bolsters a lot of camouflage, hash and encryption calculations. Alongside concealing the documents, the device can be utilized to remove the inserted records also and to finish everything off, it requires a secret phrase (that you characterize during the document concealing cycle).

2.2 Invisible Secrets 4

Invisible Secrets is a application that permits you to scramble your private documents by methods for various devices. With this program, users have the option to shield your information and messages from inquisitive eyes. The application incorporates a component that permits you to scramble and shroud documents into different record types like photographs or sound records. There's additionally another device that empowers you to secure your documents by changing them into an incoherent configuration that must be perused with the right secret key. This secret phrase is profoundly encoded utilizing solid encryption calculations.

2.3 JHide

A command line tool written in Java to shroud document records in pictures, JHide is a little programming application whose design is to help you conceal delicate documents inside custom pictures. The apparatus can be conveyed on all Windows forms out there, given that you have the Java working stage introduced on the host PC. The photos that contain private documents look equivalent to other photographs put away in your PC so they won't raise any doubts to different clients. They can be opened with similar devoted watchers, sent through email, or printed.

2.4 Open Puff

OpenPuff is a "proficient steganography tool" that permits clients to hide records in picture, sound, video, or Flash documents. It gives a wide exhibit of highlights to shield concealed information from revelation. OpenPuff is an expert steganography device with extraordinary highlights, appropriate for exceptionally delicate information undercover transmission. In OpenPuff information is part of numerous transporters. Just the right transporter succession empowers viewing. In addition, up to 256Mb can be covered up, on the off chance that you have enough transporters at removal. The last transporter will be dispatched with arbitrary pieces to make it undistinguishable from others.

2.5 OpenStego

OpenStego gives two primary functionalities: Data Hiding: It can shroud any information inside a cover document (for example pictures) and Watermarking with an undetectable mark. It tends to be utilized to recognize unapproved document duplicating. Utilizing OpenStego is quite direct. There are two methods of activity - information covering up and watermarking.

2.6 StegHide

Steghide is a steganography program that can conceal information in different sorts of picture and sound documents. The tone respectively test frequencies are not changed subsequently making the installing safe against first-request factual tests. Steghide highlights include compression of installed information, encryption of installed information, inserting of a checksum to confirm the respectability of the extracted information and backing for JPEG, BMP, WAV and AU records.

2.7 StegoShare

Stegoshare can be effortlessly utilized for unknown record sharing. An uploader downloads legitimate pictures from a public photograph facilitating webpage, and inserts the edited document into those pictures. The uploader then transfers pictures to the public photograph deluge tracker and puts the connections referring to the stego pictures with blue-penciled document's portrayal on a discussion or blog. Downloaders, seeders, and public photograph trackers, whenever found disseminating unlawful records, are shielded from lawful indictment, since they can generally utilize conceivable deniability, saying that they don't knew the slightest bit about the illegal document in the pictures.

2.8 S-Tool

S-Tools (Steganography Tools) is a program composed by Andy Brown. It is maybe the most generally perceived steganography tool accessible today. BMP, GIF, and WAV documents can be utilized as the cover records that disguise the mystery messages. It is not difficult to use, with basic relocating of the records. S-Tools will conceal the mystery message inside the cover document through arbitrary accessible pieces. These accessible pieces are resolved using a pseudorandom number generator. This nonlinear inclusion makes the presence and extraction of mystery messages more troublesome.

2.9 VSL Virtual Steganographic Laboratory

Virtual Steganographic Laboratory (VSL) is a graphical square charting tool that permits complex utilizing, testing and changing of techniques both for picture steganography and steganalysis. VSL furnishes straightforward GUI alongside secluded, module design.

2.10 Xiao Steganography

Xiao Steganography is a half breed steganography tool that permits clients to conceal documents inside pictures (BMP) or sound (WAV) records. The device likewise permits clients to scramble the shrouded record with an assortment of upheld encryption calculations (counting RC4 and 3DES) and hashing calculations (counting SHA and MD5). The client gives a transporter document (the covering for the shrouded record), the record to stow away, a decision of encryption calculation, and a secret key.

2.11 Steganography Studio

Steganography Studio software is a tool to learn, use and analyze key steganographic algorithms. It implements several algorithms highly configurable with a variety of filters. This software is developed in Java, allowing use in any operating system.

2.12 El Carpincho Project

El Carpincho Project is a Java based portable and simple program for text and file encryption (Symmetric, Asymmetric and Steganography)

2.13 Hide & Reveal

Hide & Reveal is both an open-source steganography software and a java library distributed under the GNU GPL. It is primarily designed for scientists wishing to experiment new hiding techniques or steganalysis on various carriers.

Table 1: Table of image steganography tools, source: [9, 13, 15, 17]

Steganography Tools	Brief Description	Image Formats Supported
Hide 'N' Send	Supports encryption and hashing Relies on F5 and LSB steganography algorithms	JPEG
Invisible Secrets 4	Enterprise/commercial software East-tec Corporation software product Contains email-encryption, password manager, file shredder, application locker, IP-IP password transfer and cryptoboard tool	JPEG, PNG, BMP
JHide	Simple steganography tool Java application	BMP
Open Puff	Steganography and watermarking tool Allows to embed data in more than one carrier file	BMP, JPEG, PNG, TGA
OpenStego	Open source Steganography and watermarking tool Java Application Supports encryption and password protection	BMP, PNG
StegHide	Cross platform Open source Supports compression and encryption Relies on graph theory matching algorithm	JPEG, BMP
StegoShare	Supports embedding large files in multiple image carriers Relies on fixed location LSB algorithm	BMP, JPEG, PNG, GIF, TIFF
S-Tool	Steganography tool Supports Encryption and Password protection	BMP, GIF
VSL Virtual Steganographic Laboratory	Image steganography and steganalysis software	BMP, PNG, JPEG, TIFF
Xiao Steganography	Windows platform Developed by Nakasoft Supports encryption, hashing and password protect options	BMP
Steganography Studio	Different hiding methods (LSB, LSB Matching, SLSB), Open source	BMP, PNG, GIF
El Carpincho Project	A new version, with a lot of new and great features like steganography, swing interface, more algorithms, printing, etc	JPG, JPEG, BMP, PNG, TIF, GIF, (Input); PNG (Output)
Hide & Reveal	Allows to hide any type of file within BMP, PNG and TIF images	BMP and PNG

Table 1 lists the various image steganography tools that were selected, and the list is not intended to be an exhaustive one consisting of all possible image steganography tools. Moreover, there is a filter that was applied to separate all steganography tools that were designed in Java and were Windows based with GUI interface. These filtered Java and Windows based steganography tools were tested to see the encode and decode time difference.

3 Selection of Steganography Tools to Test

The steganography tools to be tested were searched with the aid of Google Search Engine and the SourceForge site (https://sourceforge.net). The search terms used for Google Search was "java steganography tool" and the search terms used for the SourceForge site was "steganography java". The tools were then selected from the search results based on the following criteria:

- Has a GUI interface
- Can run on Windows (preferably Windows 10)
- Uses images to hide the secret message or file
- Developed using JAVA
- Not suspected to be malicious software or harboring malicious code.

A total of seven steganography tools was selected for testing when applying the above criteria:

- Jhide
- Open Stego
- StegoShare
- VSL Virtual Steganographic Laboratory
- Steganography Studio
- El Carpincho Project
- Hide & Reveal

4 Data Set Used

Image files that employ lossless compression such as BMP are considered better for steganography compared to image files that employ lossy compression such as JPEG [8, 11]. As such it is possible that one of the suspicious traits of stego images is the use of image file types that are commonly employed for steganography. Hence for the purpose of testing the seven steganography tools, it has been decided to use both BMP (lossless compression) and JPG (lossy compression) cover images.

Table 2: Table of dataset details

Steganography Dataset	Image Size	Image Format	Number of images
Steganalysis Dataset from Mendeley	512x512 px	JPEG, RGB – BMP	1500

4.1 Selection of Cover Image

A steganalysis dataset from Mendeley was used for the purpose of selecting a cover image for testing the steganography tools. From the dataset the C0002.bmp file was selected to be used. A jpg file was created from the selected cover image file using MS Paint software.

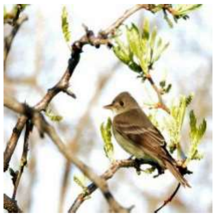


Figure 1: C0002.bmp cover image file

4.2 Selection of Secret Message Used

The secret message, "The quick brown fox jumps over the lazy dog", is stored in a textfile called Test.txt to be encoded in a cover image.

There exists a dataset, Steganalysis Dataset from Mendeley, that was specifically created for the purpose of steganography experiments. This dataset contains 3000 RGB-BMP images, dimensions 512x512, for steganography, steganalysis and similar image processing applications. It contains 1500 RGB-BMP images, dimensions 512x512, transformed from Caltech birds' dataset in JPEGC format. A detailed explanation of the dataset is described in Table 2 below.

Figure 2 represents some of the images collated from the dataset described.

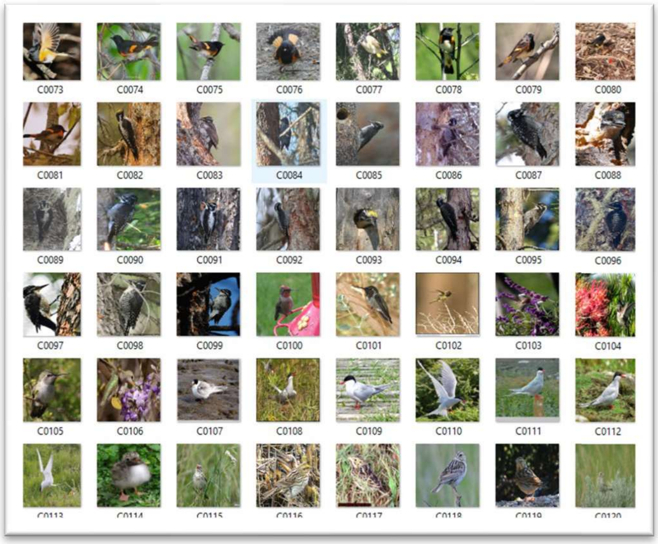


Figure 2: Table of dataset images

5 Evaluation Framework

5.1 Evaluation tool

The weighted score decision matrix was selected to evaluate the seven steganography tools as it is a commonly used in many projects for evaluating software choices [14].

The weighted score decision matrix is a project management technique that is useful in determining or selecting the most appropriate software [14]. It allows a set of choices such as in this case steganography tools to be evaluated in terms of a set of criteria that needs to be considered [5].

5.. The criteria and weightings

One of the aims of steganography is to elicit as little suspicion as possible and hence a few of the criterion will be based on structural detection as well as visual detection. Structural detection involves the comparison of the original cover image and the stego image and visual detection involves the use of the human eye to spot discrepancies in the stego image [6]. With regards to structural detection and visual detection, the following are some of the criteria to be used in the weighted score decision matrix:'

- **Visual Difference** in terms of Visual Distortion/Detectable artifacts/ Unique or unusual colors in image/ Cropping or Padding of Images [2, 10].
- **File Size Difference** [10].
- **File Type Difference** as a change in file type from cover image to stego image will yield more suspicion.
- **Numerous File Type Support** as more file type support will allow the steganography tool to potentially use image file types that are less in use in steganography such as lossy image types [8] to avoid suspicion.

The remaining criteria is related to the performance of the steganography software:

- **Encode Time** to measure time taken to encode secret message in cover image to create stego image.
- **Successful Decode Output** to see if the decoding was successful.
- **Decode Time** to measure time taken to decode secret message in stego image.

The weights for each criterion is applied in the following order:

- **Visual Difference** – weight of 30%.
- **File Size Difference** – weight of 20%.
- **File Type Difference** – weight of 10%.
- **Numerous File Type Support** – weight of 10%.
- **Encode Time** – weight of 5%.
- **Successful Decode Output** – weight of 20%.
- **Decode Time** – weight of 5%.

6 Scoring Process

The following describes how scores are assigned for each criterion. The maximum score for each criterion is 10 and the minimum is 0.

6.1 For Criteria Visual Difference, File Type Difference, Successful Decode Output

- **Visual Difference**, a score of 10 is assigned if there is no discernable visual difference between cover and stego images and a score of 0 is assigned if there is.
- **File Type Difference**, a score of 10 is assigned if there is no change in file type between cover and stego images and a score of 0 is assigned if there is.
- **Successful Decode Output**, a score of 10 is assigned if there is a successful decode output and 0 if none.

6.2 For Criteria File Size Difference, Numerous File Type Support, Encode Time, Decode Time

Scoring process for the following criteria relies on the ranking of raw readings of each test. The following will describe in detail the process of rank scoring:

- **File Size Difference,** the absolute file size difference between the cover and stego image is calculated for each test and sorted in ascending order. Tests with smaller absolute file size difference are ranked higher in comparison to tests with higher absolute file size differences. Scores are assigned based on ranks with the highest rank (rank 1) having a maximum score of 10 and the lowest rank having a score of 0.
- **Numerous File Type Support**, the number of image file types supported is determined for each test and sorted in descending order. Tests with a higher number of image file types supported are ranked higher in comparison to tests with lower number of image file types supported. Scores are assigned based on ranks with the highest rank (rank 1) having a maximum score of 10 and the lowest rank having a score of 0.
- **Encode Time**, the stopwatch app from timeanddate.com/stopwatch on google chrome on mobile phone Samsung M31 was used to record the execution time of the encode time for each test. The execution time is rounded to the nearest higher whole second and then sorted in ascending order. Tests with smaller encode times are ranked higher in comparison to tests with higher encode times. Scores are assigned based on ranks with the highest rank (rank 1) having a maximum score of 10 and the lowest rank having a score of 0.

- **Decode Time**, the stopwatch app from timeanddate.com/stopwatch on google chrome on mobile phone Samsung M31 was used to record the execution time of the decode time for each test. The execution time is rounded to the nearest higher whole second and then sorted in ascending order. Tests with smaller decode times are ranked higher in comparison to tests with higher decode times. Scores are assigned based on ranks with the highest rank (rank 1) having a maximum score of 10 and the lowest rank having a score of 0.

7 Evaluation and Experimental Setup

A test was conducted for each cover image type (jpg and bmp image files), steganography algorithm (if the steganography tool supports more than one) and the steganography tool. At least 2 tests are conducted for each tool. Each test involves an encode attempt using the specified steganography tool, specified cover image (jpg or bmp cover image file), specified steganography algorithm (if the steganography tool supports more than one), and secret message as well as a decode attempt. The output of the encoded attempt (stego image) is compared with the original cover image to see if there are any discernable differences in terms of Visual Difference, File Size Difference, and File Type Difference, with the measurements recorded in the weighted score decision matrix. The encode attempt is also timed (Encode Time) with the measurements recorded in the weighted score decision matrix. If the encode attempt is not successful, no measurements are entered in the weighted score decision matrix and the score assigned is 0 for those criteria where measurements cannot be obtained. The encode output (stego image) is then decoded with the decode attempt timed (Decode Time) and the decode attempt is checked to see if it results in a decode output or secret message (Successful Decode Output). The measurements of the decode attempt, Decode Time, and Successful Decode Output, are then entered in the weighted score decision matrix. The measurements for Numerous File Type Support can be conducted in advance of the tests and entered in the weighted score decision matrix.

The following describes the tests performed for each tool:

- Jhide, 2 tests performed: one using bmp image as cover image and the other using jpg image as cover image.
- Open Stego, 2 tests performed: one using bmp image as cover image and the other using jpg image as cover image.
- StegoShare, 2 tests performed: one using bmp image as cover image and the other using jpg image as cover image.
- VSL Virtual Steganographic Laboratory, 6 tests performed: one per image file type (jpg and bmp file) and steganography algorithm (LSB, KLT, and F5)
- Steganography Studio, 14 tests performed: one per image file type (jpg and bmp file) and steganography algorithm (BattleSteg, BlindHide, DynamicBattleSteg, DynamicFilterFirst, FilterFirst, HideSeek, and SLSB)
- El Carpincho Project, 2 tests performed: one using bmp image as cover image and the other using jpg image as cover image.
- Hide & Reveal, 6 tests performed: one per image file type (jpg and bmp file) and steganography algorithm (Single LSB, Dual LSB, and Triple LSB)

The tests were all conducted on a computer with the following specifications:

- Intel® Core™ i5-6600K CPU @ 3.50GHz
- 32GB RAM
- Windows 10 Pro 64-bit OS Version 21H1 OS build 19043.1110
- Java Version 8 Update 291 (build 1.8.0_291-b10)

8 Experiment Observation and Outcome Results

The following table figures show the experiment outcomes:

It becomes apparent from observing the before mentioned test or experiment data that several steganography tools are not able to handle lossy image files such as jpg files:

- Jhide
- Hide & Reveal

Certain steganography tools also encompass faulty steganography algorithms that may yield no encode output (stego images) such as KLT steganography algorithm from VSL Virtual Steganographic Laboratory which yields no encode output regardless of cover image type.

In terms of Visual Difference, none of the steganography tools that yielded successful encode outputs had any discernable differences when compared visually with the cover image.

With regards to File Size Difference, it would seem most of the file size difference depends on the file image type used as the cover image. Jpg image file types are more likely to yield file size differences in comparison to bmp image file types.

The file size difference may partially be explained when looking at the File Type Difference. Several steganography tools, Open Stego, StegoShare, Steganography Studio, and El Carpincha Project, create the stego image as a different image file type in comparison to the cover image file type used.

With regards to Encode Times and Decode Times of Steganography software, all successful encode and decode tests have been measured as having an execution time of 1 second or less with the sole exception of the test involving Stego Share using jpg cover image which has an execution time of 2 seconds or less for Encode Time.

Only one software, Steganography Studio, has serious issues regarding Successful Decode Output whereby all tests involving this steganography tool has no decode output (cannot retrieve the secret message). VSL Virtual Steganographic Laboratory also has no decode output only when using jpg cover image and LSB as the steganography algorithm.

When looking at the Average Total Weighted Score by

Table 3: Weighted score decision matrix for steganography tools (part 1) - visual and structural detection tests

Tools	Tests on Tools	Visual Detection/Structural Detection																
		Visual Difference				File Size Difference							File Type Difference					
		Y/N	Score	Weight	Weighted Score (3)	File Size of Cover (KB)	File Size of Stego (KB)	Difference	Absolute Value of Difference	Score	Weight	Weighted Score (2)	Image Type - Input	Image Type - Output	Same? Y/N	Score	Weight	Weighted Score (1)
			10	0.3						10	0.2					10	0.1	
Jhide	Jhide - bmp	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
Jhide	Jhide - jpg	*	0	0.3	0	*	*	*	*	0	0.2	0	jpg	*	*	0	0.1	0
Open Stego	Open Stego - jpg	N	10	0.3	3	92.4	768	-675.6	675.6	3.939393939	0.2	0.787878788	jpg	bmp	N	0	0.1	0
Open Stego	Open Stego -bmp	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
StegoShare	StegoShare - jpg	N	10	0.3	3	92.4	645	-552.6	552.6	4.545454545	0.2	0.909090909	jpg	png	N	0	0.1	0
StegoShare	StegoShare - bmp	N	10	0.3	3	768	644	124	124	5.454545455	0.2	1.090909091	bmp	png	N	0	0.1	0
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- jpg, LSB	N	10	0.3	3	92.4	166	-73.6	73.6	5.757575758	0.2	1.151515152	jpg	jpg	Y	10	0.1	1
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- jpg, KLT	*	0	0.3	0	*	*	*	*	0	0.2	0	jpg	*	*	0	0.1	0
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- jpg, F5	N	10	0.3	3	92.4	160	-67.6	67.6	6.060606061	0.2	1.212121212	jpg	jpg	Y	10	0.1	1
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- bmp, LSB	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- bmp, KLT	*	0	0.3	0	*	*	*	*	0	0.2	0	bmp	*	*	0	0.1	0
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- bmp, F5	N	10	0.3	3	768	194	574	574	4.242424242	0.2	0.848484848	bmp	jpg	N	0	0.1	0
Steganography Studio	Steganography Studio - jpg, BattleSteg	N	10	0.3	3	92.4	768	-675.6	675.6	3.939393939	0.2	0.787878788	jpg	bmp	N	0	0.1	0
Steganography Studio	Steganography Studio - jpg, BlindHide	N	10	0.3	3	92.4	768	-675.6	675.6	3.939393939	0.2	0.787878788	jpg	bmp	N	0	0.1	0
Steganography Studio	Steganography Studio - jpg, DynamicBattleSteg	N	10	0.3	3	92.4	768	-675.6	675.6	3.939393939	0.2	0.787878788	jpg	bmp	N	0	0.1	0
Steganography Studio	Steganography Studio - jpg, DynamicFilterFirst	N	10	0.3	3	92.4	768	-675.6	675.6	3.939393939	0.2	0.787878788	jpg	bmp	N	0	0.1	0
Steganography Studio	Steganography Studio - jpg, FilterFirst	N	10	0.3	3	92.4	768	-675.6	675.6	3.939393939	0.2	0.787878788	jpg	bmp	N	0	0.1	0

Steganography Studio	Steganography Studio - jpg, HideSeek	N	10	0.3	3	92.4	768	-675.6	675.6	3.939393939	0.2	0.787878788	jpg	bmp	N	0	0.1	0
Steganography Studio	Steganography Studio - jpg, SLSB	N	10	0.3	3	92.4	768	-675.6	675.6	3.939393939	0.2	0.787878788	jpg	bmp	N	0	0.1	0
Steganography Studio	Steganography Studio - bmp, BattleSteg	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
Steganography Studio	Steganography Studio - bmp, BlindHide	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
Steganography Studio	Steganography Studio - bmp, DynamicBattleSteg	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
Steganography Studio	Steganography Studio - bmp, DynamicFilterFirst	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
Steganography Studio	Steganography Studio - bmp, FilterFirst	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
Steganography Studio	Steganography Studio - bmp, HideSeek	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
Steganography Studio	Steganography Studio - bmp, SLSB	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
El Carpincho Project	El Carpincho Project - jpg	N	10	0.3	3	92.4	536	-443.6	443.6	4.848484848	0.2	0.96969697	jpg	png	N	10	0.1	1
El Carpincho Project	El Carpincho Project - bmp	N	10	0.3	3	768	572	196	196	5.151515152	0.2	1.03030303	bmp	png	N	10	0.1	1
Hide & Reveal	Hide & Reveal - bmp, Single LSB	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
Hide & Reveal	Hide & Reveal - bmp, Dual LSB	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
Hide & Reveal	Hide & Reveal - bmp, Triple LSB	N	10	0.3	3	768	768	0	0	10	0.2	2	bmp	bmp	Y	10	0.1	1
Hide & Reveal	Hide & Reveal - jpg, Single LSB	*	0	0.3	0	*	*	*	*	0	0.2	0	jpg	*	*	0	0.1	0
Hide & Reveal	Hide & Reveal - jpg, Dual LSB	*	0	0.3	0	*	*	*	*	0	0.2	0	jpg	*	*	0	0.1	0
Hide & Reveal	Hide & Reveal - jpg, Triple LSB	*	0	0.3	0	*	*	*	*	0	0.2	0	jpg	*	*	0	0.1	0

* Implies data that cannot be recorded

Table 4: Weighted score decision matrix for steganography tools (part 2) - features and performance tests of steganography tools as well as total weighted score

Tools	Tests on Tools	Features and Performance of Steganography Tools																		
		Numerous File Type Support				Encode Time					Successful Decode Output				Decode Time					
		Number of File Types supported	Score	Weight	Weighted Score (1)	Approximate Time	Time Range	Score	Weight	Weighted Score (0.5)	Y/N	Score	Weight	Weighted Score (2)	Approximate Time	Time Range	Score	Weight	Weighted Score (0.5)	Total Weighted Score (10)
			10	0.1				10	0.05			10	0.2				10	0.05		
Jhide	Jhide - bmp	4	6.36363636	0	0.636364	0.551	1	10	0.05	0.5	Y	10	0.2	2	0.386	1	10	0.05	0.5	9.636363636
Jhide	Jhide - jpg	4	6.36363636	0	0.636364	*	*	0	0.05	0	N	0	0.2	0	*	*	0	0.05	0	0.636363636
Open Stego	Open Stego - jpg,	6	10	0	1	1.076	2	1.8182	0.05	0.090909091	Y	10	0.2	2	0.927	1	10	0.05	0.5	7.378787879
Open Stego	Open Stego -bmp,	6	10	0	1	0.459	1	10	0.05	0.5	Y	10	0.2	2	0.491	1	10	0.05	0.5	10
StegoShare	StegoShare - jpg	5	6.96969697	0	0.69697	0.77	1	10	0.05	0.5	Y	10	0.2	2	0.647	1	10	0.05	0.5	7.606060606
StegoShare	StegoShare - bmp	5	6.96969697	0	0.69697	0.462	1	10	0.05	0.5	Y	10	0.2	2	0.709	1	10	0.05	0.5	7.787878788
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- jpg, LSB	6	10	0	1	0.368	1	10	0.05	0.5	N	0	0.2	0	*	*	0	0.05	0	6.651515152
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- jpg, KLT	6	10	0	1	*	*	0	0.05	0	N	0	0.2	0	*	*	0	0.05	0	1
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- jpg, F5	6	10	0	1	0.395	1	10	0.05	0.5	Y	10	0.2	2	0.541	1	10	0.05	0.5	9.212121212
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- bmp, LSB	6	10	0	1	0.454	1	10	0.05	0.5	Y	10	0.2	2	0.533	1	10	0.05	0.5	10
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- bmp, KLT	6	10	0	1	*	*	0	0.05	0	N	0	0.2	0	*	*	0	0.05	0	1
VSL Virtual Steganographic Laboratory	VSL Virtual Steganographic Laboratory- bmp, F5	6	10	0	1	0.484	1	10	0.05	0.5	Y	10	0.2	2	0.408	1	10	0.05	0.5	7.848484848
Steganography Studio	Steganography Studio - jpg, BattleSteg	4	6.36363636	0	0.636364	0.52	1	10	0.05	0.5	N	0	0.2	0	0.627	1	10	0.05	0.5	5.424242424
Steganography Studio	Steganography Studio - jpg, BlindHide	4	6.36363636	0	0.636364	0.498	1	10	0.05	0.5	N	0	0.2	0	0.516	1	10	0.05	0.5	5.424242424
Steganography Studio	Steganography Studio - jpg, DynamicBattleSteg	4	6.36363636	0	0.636364	0.87	1	10	0.05	0.5	N	0	0.2	0	0.765	1	10	0.05	0.5	5.424242424

Steganography Studio	Steganography Studio - jpg, DynamicFilterFirst	4	6.36363636	0	0.636364	0.566	1	10	0.05	0.5	N	0	0.2	0	0.585	1	10	0.05	0.5	5.424242424
Steganography Studio	Steganography Studio - jpg, FilterFirst	4	6.36363636	0	0.636364	0.65	1	10	0.05	0.5	N	0	0.2	0	0.622	1	10	0.05	0.5	5.424242424
Steganography Studio	Steganography Studio - jpg, HideSeek	4	6.36363636	0	0.636364	0.825	1	10	0.05	0.5	N	0	0.2	0	0.534	1	10	0.05	0.5	5.424242424
Steganography Studio	Steganography Studio - jpg, SLSB	4	6.36363636	0	0.636364	0.576	1	10	0.05	0.5	N	0	0.2	0	0.628	1	10	0.05	0.5	5.424242424
Steganography Studio	Steganography Studio - bmp, BattleSteg	4	6.36363636	0	0.636364	0.689	1	10	0.05	0.5	N	0	0.2	0	0.721	1	10	0.05	0.5	7.636363636
Steganography Studio	Steganography Studio - bmp, BlindHide	4	6.36363636	0	0.636364	0.683	1	10	0.05	0.5	N	0	0.2	0	0.645	1	10	0.05	0.5	7.636363636
Steganography Studio	Steganography Studio - bmp, DynamicBattleSteg	4	6.36363636	0	0.636364	0.505	1	10	0.05	0.5	N	0	0.2	0	0.974	1	10	0.05	0.5	7.636363636
Steganography Studio	Steganography Studio - bmp, DynamicFilterFirst	4	6.36363636	0	0.636364	0.671	1	10	0.05	0.5	N	0	0.2	0	0.616	1	10	0.05	0.5	7.636363636
Steganography Studio	Steganography Studio - bmp, FilterFirst	4	6.36363636	0	0.636364	0.85	1	10	0.05	0.5	N	0	0.2	0	0.715	1	10	0.05	0.5	7.636363636
Steganography Studio	Steganography Studio - bmp, HideSeek	4	6.36363636	0	0.636364	0.534	1	10	0.05	0.5	N	0	0.2	0	0.572	1	10	0.05	0.5	7.636363636
Steganography Studio	Steganography Studio - bmp, SLSB	4	6.36363636	0	0.636364	0.605	1	10	0.05	0.5	N	0	0.2	0	0.559	1	10	0.05	0.5	7.636363636
El Carpincho Project	El Carpincho Project - jpg	6	10	0	1	0.479	1	10	0.05	0.5	Y	10	0.2	2	0.346	1	10	0.05	0.5	8.96969697
El Carpincho Project	El Carpincho Project - bmp	6	10	0	1	0.55	1	10	0.05	0.5	Y	10	0.2	2	0.408	1	10	0.05	0.5	9.03030303
Hide & Reveal	Hide & Reveal - bmp, Single LSB	3	1.51515152	0	0.151515	0.973	1	10	0.05	0.5	Y	10	0.2	2	0.516	1	10	0.05	0.5	9.151515152
Hide & Reveal	Hide & Reveal - bmp, Dual LSB	3	1.51515152	0	0.151515	0.386	1	10	0.05	0.5	Y	10	0.2	2	0.456	1	10	0.05	0.5	9.151515152
Hide & Reveal	Hide & Reveal - bmp, Triple LSB	3	1.51515152	0	0.151515	0.554	1	10	0.05	0.5	Y	10	0.2	2	0.435	1	10	0.05	0.5	9.151515152
Hide & Reveal	Hide & Reveal - jpg, Single LSB	3	1.51515152	0	0.151515	*	*	0	0.05	0	N	0	0.2	0	*	*	0	0.05	0	0.151515152
Hide & Reveal	Hide & Reveal - jpg, Dual LSB	3	1.51515152	0	0.151515	*	*	0	0.05	0	N	0	0.2	0	*	*	0	0.05	0	0.151515152
Hide & Reveal	Hide & Reveal - jpg, Triple LSB	3	1.51515152	0	0.151515	*	*	0	0.05	0	N	0	0.2	0	*	*	0	0.05	0	0.151515152

* Implies data that cannot be recorded

Table 5: Average total weighted score by steganography tool

Tools	Average Total Weighted Score (10)
Jhide	5.1363636
Open Stego	8.6893939
StegoShare	7.6969697
VSL Virtual Steganographic Laboratory	5.9520202
Steganography Studio	6.530303
El Carpincho Project	9
Hide & Reveal	4.6515152

Steganography Tools, the top 3 steganography tool with the highest average total weighted score is the following:

1. El Carpincho Project
2. Open Stego
3. StegoShare

The bottom 3 steganography tool with the lowest average total weighted score is the following:

1. Hide & Reveal
2. Jhide
3. VSL Virtual Steganographic Laboratory

9 Challenges

Finding tools and techniques in computer forensic investigations to decipher the information hidden in steganography when necessary is challenging. It is not only to detect the existence of steganography but is significantly necessary to reveal hidden data information. [3] The development of this general tool for the detection of steganography and classification is still under development. Certain criteria are difficult to measure with sufficient accuracy:

- Measuring Encode Time and Decode Time with sufficient accuracy is difficult due to the manual nature of timing the execution.
- Discerning Visual Difference is subjective depending on the person observing the difference between Cover and Stego images.

There is a slight risk that certain steganography tools that could have met the necessary selection criteria have been overlooked and not included in the experiment.

10 Future Works and Discussion

The functioning rule of image steganography is to hide information in encrypted picture and link them to deliver a final product nearer to the original picture. [12] With the increasing application of steganography in the digital world, many issues must be understood in computer forensic inspection. There is a wide variety of tools and techniques, with their own focus and weaknesses. Stable changes must be made and more recent adaptations have been made. Initially, steganography overview provided computer forensics experts in this field with knowledge and understanding of steganography. A portion of the viewpoints that can be considered for future works are specified herein. Most of image steganography techniques use pictures as the privileged intel and there is a requirement for more examination secluded from text in picture. Analyses identified with upgrading the boundaries and diminishing the capacity limits can be additionally directed utilizing different datasets. The use of tools for observing bit changes in data can also raise suspicion, as investigated. The procedures and algorithms used in steganography are analyzed to be used as the basis for understanding how steganography works. The Image file format is the most widely used digital media medium. [16] Endeavors can be coordinated to shape a benchmark dataset containing pictures from different source cameras and picture designs. An assemblage of all potential calculations should likewise be possible to make the steganography pictures. “

11 Conclusion

It tends to be plainly seen that many picture steganography tools have blemishes that take into consideration the simple recognition or sign of a concealed document in a stego-picture. As such it is basic that individuals engaged with data security use those picture steganography tools that will diminish the opportunity of the message being identified since the strength of steganography is not to make the message ambiguous but instead to cause it to appear as though it was never there in any case. Four devices were discovered to be such devices that will limit the opportunity of outsiders recognizing a shrouded message in the stego-picture. As expressed before this paper is not proposed to be a thorough request and test into all conceivable image-based steganography tools and it is conceivable to additionally improve the weighted score framework used to evaluate the steganography instruments by fusing extra boundaries that may address other significant aspects of steganography, for example, strength.

Many image steganography tools that were randomly selected and the list is not intended to be an exhaustive one consisting of all possible image steganography tools. Additionally, there is a filter that was applied to separate all steganography tools that were designed in Java and were windows based with GUI interface. These filtered Java and Windows based steganography tools were tested to see the encode and decode time difference, the visual differences and file size variance. In any case, the authors place that this paper will be helpful for those specialists meaning to use similar basic pointers of concealed information and those expecting to test, survey and use picture steganography frameworks.

This paper began with a brief overview of the steganography tools to be reviewed, the tests to perform on such tools and the analysis of the results that helped determine which of the image steganography tools will yield stego-images that are nondescript in nature. In synopsis, this paper has expounded on the methods utilized in the new occasions for picture steganography, the latest things. Lastly, this research provides clues for computer forensic inspectors to understand the importance of type steganography tools installed, hidden, or removed on victims' computers. Finding evidence of the suspect, a certain steganography tool will trigger a suspicious impression. As shown in the results of the experiment, it is necessary to know the type of steganography tool. It very well may be inferred that deep learning has enormous potential in the picture steganography field contemplating that every one of the difficulties and challenges are filled.

References

[1] A. Aljarf, S. Amin and J. Filippas, “Creating Stego-Images Through Hiding Single and Multipile Data Using Different Steganographic Tools,” *Proceedings of the IASTED International Conference*, Innsbruck, pp. 343-348, 2013.

[2] A. Almohammad and G. Ghinea, “Stego Image Quality and the Reliability of PSNR,” 2010 2nd International Conference on Image Processing Theory, Tools and Applications, Paris, 2010.

[3] J. Butora and J. Fridrich, “Effect of JPEG Quality on Steganographic Security,” *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security*, 2019.

[4] T. Denemark, P. Bas, and J. Fridrich, “Natural Steganography in JPEG Compressed Images,” *Electronic Imaging,* 7:1-10, 2018.

[5] Airfocus GmbH, “Weighted Scoring,” [Online]. Available: https://airfocus.com/guides/prioritization/7-most-popular-prioritization-frameworks/weighted-scoring/, [Accessed 2 September 2021].

[6] N. A. Hassan and R. Hijazi, *Data Hiding Techniques in Windows OS*, Syngress, 2017.

[7] A. Kaur, R. Kaur, and N. Kumar, “A Review on Image Steganography Techniques,” *International Journal of Computer Applications,* 123(4):20-24, 2015.

[8] G. C. Kessler, “Steganography: Hiding Data Within Data,” September 2001, [Online], Available: https://www.garykessler.net/library/steganography.html. [Accessed 1 September 2021.

[9] V. Kumar, “30 Useful Steganography Tools To Hide Secret Information,” 2 September 2017, [Online], Available: https://www.rankred.com/steganography-tools-to-hide-secret-information/. [Accessed 17 October 2019].

[10] A. Kumar and K. Pooja, “Steganography - A Data Hiding Technique,” *International Journal of Computer Applications,* 9(7):19-23, 2010.

[11] X. Liao, J. Yin, S. Guo, X. Li, and A. K. Sangaiah, “Medical JPEG Image Steganography Based on Preserving Inter-Block Dependencies,” *Computers & Electrical Engineering,* 67:320-329, 2018.

[12] L. Liu, Z. Wang, Z. Qian, X. Zhang, and G. Feng, “Steganography in Beautified Images,” *Mathematical Biosciences and Engineering,* 16(4):2322-2333, 2019.

[13] C. Maji, “Describing Textures in the Wild,” *IEEE Conf. on Computer Vision and Pattern Recognition*, 2014.

[14] N. Morpus, “A Step-by-Step Guide for Using a Weighted Scoring Model,” 2 September 2021. [Online], Available: https://www.fool.com/the-blueprint/weighted-scoring-model/, [Accessed 16 January 2021].

[15] H. Passi, “Top 10 Must-Have Tools to Perform Steganography,” GreyCampus, 5 October 2018, [Online], Available: https://www.greycampus.com/blog/information-security/top-must-have-tools-to-perform-steganography, [Accessed 17 October 2019].

[16] D. R. I. M. Setiadi, E. H. Rachmawanto, and C. A. Sari, “Secure Image Steganography Algorithm Based on DCT with OTP Encryption,” *Journal of Applied Intelligent System,* 2(1):1-11, 2017.

[17] P. Shankdhar, “Best Tools to Perform Steganography [Updated 2019],” 17 May 2019, [Online], Available: https://resources.infosecinstitute.com/steganography-and-tools-to-perform-steganography/#gref, [Accessed 17 October 2019].

[18] V. Sharma and S. Kumar, “A New Approach to Hide Text in Images Using Steganography,” *International Journal of Advanced Research in Computer Science and Software Engineering,* III(4):701-708, 2013.

[19] A. Siper, R. Farley, and C. Lombardo, “The Rise of Steganography,” 6 May 2005, [Online], Available: http://csis.pace.edu/~ctappert/srd2005/d1.pdf, [Accessed 17 October 2019].

[20] M. E. Whitman and H. J. Mattord, *Principles of Information Security*, 5th ed., Boston: Cengage Learning, 2014.

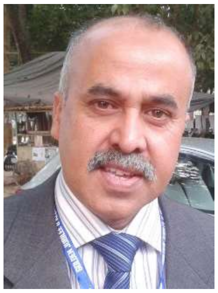
Anal Kumar was awarded a BIT degree from the University of Fiji in 2009 and Master of Science in Information Technology in 2016. He is currently a Lecturer at Department of Computing Sciences and Information Systems at Fiji National University and pursuing PhD in

Information Technology through the University of Fiji. E-mail: anal.kumar@fnu.ac.fj

Hermann Jamnadas completed his Bachelor of Commerce in Accounting and Information Systems from The University of the South Pacific, Laucala, Fiji in the year of 2011. He obtained his Postgraduate Diploma in University of Fiji, Saweni, Fiji in the year 2016 as well as his Master of Information Technology from The University of Fiji, Information Technology from The Saweni, Fiji in the year 2022. E-mail: hermann.jamnadas@fnu.ac.fj

Vishal Sharma completed his Masters of Computing Science and Information in 2013 from the University of the South Pacific, along with other IT certifications such as CCNA, CISSP and Professional Training in Big Data Analytics. He is an academic with almost 15 years of experience and has research interest in areas of Networking, Mobile Commerce, Big data Analytics, Cybersecurity. E-mail: vishal.sharma@fnu.ac.fj

S.M Muyeen is presently working as a Professor in the Electrical Engineering Department of Qatar University, Doha, Qatar. He received his Ph.D. from the Kitami Institute of Technology, Japan, in Electrical and Electronic Engineering and worked in Japan under the versatile banner of the Japan Society for the Promotion of Science (JSPS). He holds visiting/adjunct positions with many foreign universities, e.g., Shanghai Maritime University, China, Curtin University, Australia, Shanghai University of Electric Power, China. E-mail: ieee.csde@gmail.com



ABM Shawkat Ali is a Bangladeshi origin-Australian author, computer scientist and data analyst. He is the author of several books in the area of Data Mining, Computational Intelligence, and Smart Grid. He is a newspaper columnist. He is an academic and well-known researcher in the areas of Machine Learning and Data Science. He is also the founder of a research center and international conferences in Data Science and Engineering. He is now a Professor in Data Science at the University of Fiji. E-mail: abm.shawkat.ali@gmail.com