KAUNAS UNIVERSITY OF TECHNOLOGY

INFORMATICS FACULTY

Kalki Kumar Ballampalli

Steganography using Surface Amalgamation

Master's Degree Final Project

Supervisor

(signature) Assoc. prof. dr. Tomas Blazauskas

KAUNAS UNIVERSITY OF TECHNOLOGY

INFORMATICS FACULTY

Steganography using Surface Amalgamation

Master's DegreeFinal Project

Project made by

(signature) Kalki Kumar Ballampalli

(date)

Final Degree Project (code P000M106)

Supervisor

(signature) Assoc. prof. dr. Tomas Blazauskas

(date)

Reviewer

(signature)

(date)

KAUNAS,2016

KAUNAS UNIVERSITY OF TECHNOLOGY

Informatics Faculty

(Faculty)

Kalki Kumar Ballampalli

(Student's name, surname)

Final Degree Project (P000M106)

(Title and code of study program)

Steganography Using Surface Amalgamation

DECLARATION OF ACADEMIC INTEGRITY

____ ________________ 20 ____

Kaunas

I confirm that the final project of mine, **Kalki Kumar Ballampalli**, on the subject "Steganography Using Surface Amalgamation" is written completely by myself; all the provided data and research results are correct and have been obtained honestly. None of the parts of this thesis have been plagiarized from any printed, Internet-based or otherwise recorded sources. All direct and indirect quotations from external resources are indicated in the list of references. No monetary funds (unless required by law) have been paid to anyone for any contribution to this thesis.

I fully and completely understand that any discovery of any manifestations/case/facts of dishonesty inevitably results in me incurring a penalty according to the procedure(s) effective at Kaunas University of Technology.

(name and surname filled in by hand) *(signature)*

Kalki Kumar Ballampalli. Steganography Using Surface Amalgamation. *Master's*FinalProject / supervisorassoc.prof. dr.Tomas Blazauskas; Faculty of Informatics , Kaunas University of Technology.

Research field and area: Physical sciences / Informatics

Keywords:

Kaunas, 2016. 05 p.

Kalki Kumar Ballampalli. Steganography Using Surface Amalgamation.Master'sFinal Project / supervisorassoc.prof. dr.Tomas Blazauskas; Faculty of Informatics , Kaunas University of Technology.

Research field and area: Physical sciences / Informatics

Keywords:

Kaunas, 2016. 05 p.

SUMMARY

I propose a novel methodology for steganography utilizing a surface amalgamation procedure. A composition union procedure resamples a littler surface picture, which combines another surface picture with a comparative nearby appearance and a self-assertive size. We mesh the surface combination process into steganography to cover mystery messages.

Rather than utilizing a current spread picture to cover messages, our calculation disguises the source surface picture and implants mystery messages through the procedure of composition combination. This permits us to extricate the mystery messages and source surface from a stego manufactured composition.

Our methodology offers three unmistakable points of interest. Initially, our plan offers the installing limit that is corresponding to the extent of the stego composition picture. Second, a steganalytic calculation is not liable to crush our steganographic approach. Third, the reversible ability acquired from our plan gives usefulness, which permits recuperation of the source composition. Exploratory results have checked that our proposed approach can give different quantities of implanting limits, deliver an outwardly conceivable surface pictures, and recuperate the source composition.

Table of Contents

Abbreviations

AES	Advanced Encryption Scheme
RSA	Ron Rivest, Adi Shamir and Leonard Adleman
DSA	Digital Signature Algorithm
DES	Data Encryption Standard
HTML	Hyper Text Markup Language
J2EE	Java 2 Enterprise Edition
JAVA	A ProgrammingLanguage
JRE	Java RuntimeEnvironment
JSP	Java Server Page
JVM	Java Virtual Machine
SQL	Structured Query Language
XML	Extensible Markup Language

INTRODUCTION:

Steganography is the investigation of implanting and concealing messages in a medium called a cover text [1]. Steganography is identified with cryptography and is just about as old. It was utilised by the Ancient Greeks to shroud data about troop developments by tattooing the data on somebody's head and afterwards giving the individual a chance to become out their hair. Basically, Steganography is as old as the earth. Steganography is a solitary strategy for data concealing procedures [2]

A common place Steganography application incorporates incognito interchanges between two gatherings whose presence is obscure to a conceivable assailant and whose achievement relies on upon recognising the existence of this correspondence.Steganography is utilising reversible composition combination[3] [4]. A combination composition process re-tests a little surface picture drawn by a craftsman or caught in a photo keeping in mind the end goal to orchestrate another structure picture with a similar neighbourhood appearance and subjective size. We mesh the surface blend process into Steganography covering mystery messages and also the source composition. Specifically, as opposed to utilising a current spread picture to shroud messages, our calculation comprises the composition of origin picture and implants mystery messages through the procedure of surface amalgamation. This permits us to separate the mystery messages and the source surface from a stego engineered composition [5].

To the best of our insight, Steganography exploiting the reversibility has ever been introduced inside the writing of composition blend. Initially, since the surface combination can combine a subjective size of composition pictures, the instaling limit which our plan offers is about the span of the stego surface picture [6]. Besides, a steganalytic calculation is not liable to thrashing this Steganography approach following the stego surface picture is made out of a source composition as opposed to by adjusting the current picture substance. Third, the reversible ability acquired from our plan gives usefulness to recuperate the structure of origin. Since the recouped source surface is precisely the same as the first source composition, it can be utilised to continue onto the second round of mystery messages for Steganography if necessary. Exploratory results have confirmed that our proposed calculation can give different quantities of inserting limits, create outwardly conceivable surface pictures, and recuperate the source composition. Hypothetical examination shows that there is an unimportant likelihood of separating our Steganography approach, and the plan can oppose an RS steganalysis assault [7] [8].

1.1.Aim and Objectives

To perform steganography in a more efficient and Productive manner unlike the approaches which are available like LSB and a many more etc.

1. To perform analysis of the existing approaches available to perform Steganography.
2. To design and implement a new steganographic approach using surface amalgamation and reverse texture synthesis.

.

WORK OBJECT:

The existing system of the greater part of the Steganography is LSB Algorithm. This implies Least Significant Algorithm [9]. In LSB calculation, the message bit is taken from the message byte and afterwards that specific bit will be installed inside the minimum huge piece of a picture or video or sound record. The message implanted at all huge piece of a picture document won't draw the suspicion of the programmer as the moment contrast that would be made in the pixel estimation of the picture record won't be seen by the ordinary stripped human eye [10]. The message that will be inserted in the LSB of a sound document won't make suspicion to the programmer as that change would not be seen by the human ear. The same idea works out even with video document.

In any case, there are a couple of shortcomings of utilising LSB. It is extremely touchy to any sifting or control of the stego-picture.Scaling, pivot, editing, expansion of clamour, or lossy pressure to the stego-picture will wreck the message [11] [12]. Then again, for the concealing limit, the extent of data to be shrouded moderately depends on to the measure of the spread picture. The message size must be littler than the picture. A large limit permits the utilisation of the littler spread picture for the message of settled size, and along these lines diminishes the data transfer capacity required to transmit the stego-picture. Another shortcoming is an aggressor can without much of a stretch destruct the message by expelling or focusing the whole LSB plane with almost no adjustment in the perceptual nature of the altered stego-picture[13].Therefore, on the off chance that this technique causes somebody to suspect something covered up in the stego-picture, then the strategy is not achievement.

DISADVANTAGE

- ✓ Low vigor to malicious assaults.
- ✓ Vulnerable to unintentional or ecological noise.
- ✓ Low temper resistance.

PROPOSED SYSTEM:

In proposed, we utilise DES (Data Encryption Standard), Triple DES (Triple Data Encryption Standard), RSA (Rivest-Shamir-Adleman) Algorithms to install the information. [14] [15] [16]. These calculations are superior to anything LSB Algorithms. An AVI (Audio Video Interleave) fie is only a grouping of great determination picture called outlines. It is conceivable to gather all the casings in bitmap design. Every housing is comprising of three channel of RGB[17]. In the wake of gathering the casing, we perform DCT (8x8 square) on any channel (say R channel) of the edges and install the mystery data bits in chose higher request coefficients. Every corner is handled by 8x8 Inverse DCT piece preparing and the joined to get AVI with the hided message.

Translating is done the backward procedure of encoding. To begin with every edge is separated from just made AVI. Perform 8x8 DCT square handling on the channel where mystery data was installed before (R-channel here) and mystery bit data's are separated by subtracting from unique DCT piece prepared qualities[18].

ADVANTAGE

- ✓ Resistance to bruteforce assaults.
- ✓ Eliminate Security Issues.
- ✓ User Friendly

2. Analysis

2.1. MINIMIZING ADDITIVE DISTORTION IN STEGANOGRAPHY USING SYNDROME-TRELLIS CODES(2011)

This paper proposes a complete down to earth philosophy for minimizing added substance contortion in Steganography with general (nonbinary) installing operation [98]. Let each conceivable estimation of each stego component be appointed a scalar communicating the twisting of an implanting change done by supplanting the spread component by this worth. The aggregate contortion is thought to be a total of per-component mutilations. Both the payload-restricted sender (minimizing the aggregate contortion while installing an altered payload) and the twisting constrained sender (boosting the payload while presenting a settled aggregate mutilation) are considered. With no loss of execution, the nonbinary case is decayed into a few double cases by supplanting singular bits in spread components. The double case is drawn nearer utilizing a novel disorder coding plan in light of double convolutional codes furnished with the Viterbi calculation [99]. This quick and extremely flexible arrangement accomplishes cutting edge results in Steganography applications while having straight time and space multifaceted nature w.r.t. the quantity of spread components. We report broad exploratory results for an expansive arrangement of relative payloads and for various mutilation profiles, including the wet paper channel. Functional value of this methodology is accepted by developing and testing versatile inserting plans for advanced pictures in raster and change spaces. Most present coding plans utilized as a part of Steganography (lattice inserting, wet paper codes, and so on.) and numerous new ones can be actualized utilizing this system [100].

2.2. F5 – A STEGNOGRAPHY ALGORITHM (2001)

Numerous Steganography frameworks are frail against visual and measurable assaults. Frameworks without these shortcomings offer just a moderately little limit for Steganography messages. The recently created calculation F5 withstands visual and factual assaults, yet regardless it offers a vast Steganography limit. F5 actualizes framework encoding to enhance the proficiency of installing. In this manner it lessens the quantity of important changes[101]. F5 utilizes permutative straddling to consistently spread out the progressions over the entire steganogram[102][103]

2.3. STATISTICALLY UNDETECTABLE JPEG STEGNOGRAPHY. DEAD END CHALLENGES AND OPPENTUNITIES (2007)

The objective of this paper is to decide the Steganography limit of JPEG pictures (the biggest payload that can be imperceptibly implanted) as for current best steganalytic techniques[104]. Moreover, by testing chose Steganography calculations we assess the impact of particular configuration components and standards, for example, the decision of the JPEG compressor, framework installing, versatile substance subordinate determination channels, and insignificant mutilation Steganography utilizing side data at the sender. From our analyses, we infer that the normal Steganography limit of grayscale JPEG pictures with quality variable 70 is roughly 0.05 bits for every non-zero AC DCT coefficient.

2.4. MODIFIED MATRIX ENCODING TECHNIQUE FOR MINIMAL DISTORTION STEGNOGRAPHY(2006)

It is surely understood that all data concealing techniques that adjust the slightest huge bits bring bends into the spread items[105]. Those twists have been used by steganalysis calculations to distinguish that the items had been adjusted. It has been suggested that just coefficients whose change does not present huge mutilations ought to be utilized for inserting. In this paper we propose a proficient calculation for data stowing away in the LSBs of JPEG coefficients. Our calculation utilizes changed framework encoding to pick the coefficients whose adjustments present insignificant implanting bending. We infer the normal estimation of the implanting mutilation as an element of the message length and the likelihood dissemination of the JPEG quantization blunders of spread pictures. Our investigations indicate close assention between the hypothetical forecast and the genuine inserting mutilation[106]. Our calculation can be utilized for both Steganography and delicate watermarking and in addition in different applications in which it is important to keep the mutilation as low as could be allowed[107].

2.5.IMAGE RETRIEVAL WITH INTERACTIVE QUERY DESCRIPTION AND DATABASE REVISION (2012)

Pictures are the most straightforward and most ideal method for representation of thoughts. The importance of pictures has been extensively expanded by the site pages. In this manner proficient picture recovery frameworks are key. Content-based picture recovery (CBIR) frameworks are the most recent territory of examination. In this paper, a shrewd picture recovery framework in view of a novel strategy called database amendment (DR) is proposed[108]. Picture highlight extraction as far as shading, composition and shape is utilized to recover pictures from the database. The consequence of highlight comparability examination of the question picture with database pictures modifies the database. The framework is made intelligent for the clients to distinguish the pictures that are most fulfilled to the need. The client fulfilled pictures are dissected and the database is reexamined to make the framework keen. Test results and examinations are introduced to show the plausibility of the proposed strategy[109]

2.6. A HUMAN-ORIENTED IMAGE RETRIEVAL SYSTEM USING INTERACTIVE GENETIC ALGORITHM (2002)

This paper gives an outline thought of recovering pictures from an expansive database. CBIR is utilized for programmed indexing and recovery of pictures relying on substance of pictures known as elements[110]. The components might be low level or High level. The low level components incorporate shading, surface and shape. The abnormal state highlight portrays the idea of human cerebrum. The contrast between low level elements separated from pictures and the abnormal state data need of the client known as semantic hole. A Single component can speak to just part of the picture property. So numerous components are utilized to upgrade the picture recovery process[111][112]. This paper has utilized shading histogram, shading mean, shading structure descriptor and composition for highlight extraction[113]. The element coordinating technique depends on their Euclidean separation.

INPUT DESIGN

The Input design is the connection between the data framework and the client. It involves the creating particular and methods for information readiness and those strides are necessary to put exchange information into a usable structure for handling can be accomplished by examining the PC to peruse information from a composed or printed record or it can happen by having individuals entering the information specifically into the framework [23]. The configuration of data spotlights on controlling the measure of information required, controlling the blunders, evading delay, dodging additional steps and keeping the procedure straightforward 24]. The data is outlined in such a path in this way, to the point that it gives security and usability withholding the protection. Information Design considered the accompanying things:

What information ought to be given as data?

How the information ought to be arranged or coded?

The exchange to manage the working staff in delivering data.

Methods for getting ready info acceptances and ventures to take after when blunder happen.

3. Objectives

1. Input Design is the procedure of changing over a client arranged portrayal of the data into a PC-based framework. This outline is essential to avoid errors in the information data process and demonstrate the right heading to the administration for getting good data from the automated framework [25].
2. It is accomplished by making easy to understand screens for the information section to handle vast volume of information. The objective of planning info is to make information section less demanding and to be free from errors [26] [27]. The data entry screen is composed in a manner that every one of the information controls can be performed. It likewise gives record seeing offices.
3. At the point when the information is entered, it will check for its validity. Information can be registered with the assistance of screens[28]. Suitable messages are given as when required so that the client won't be in maize of moment. In this way, the target of input design outline is to make a data design that is anything but difficult to take after.

4. OUTPUT DESIGN

A quality output is one, which meets the requirements of the end client and presents the data apparently. In any framework consequences of handling are imparted to the customers and other structure through yields[29]. In output design, it is resolved how the data is to be displaced for immediate need Furthermore the printed copy return. It is the most vital and direct source data to the client. Proficient and shrewd yield plan enhances the framework's relationship to help central client leadership [30].

1. Designing PC return ought to continue in a sorted out, well thoroughly considered way; the right yield must be produced while guaranteeing that every yield component is planned so individuals will discover the framework can utilise effortlessly and adequately[31]. At the point when investigation expects PC return, they ought to Identify the particular yield that is expected to meet the necessities.
2. Select strategies for exhibiting data.
3. Make record, report, or different organisations that contain data created by the framework.

The output type of a data context ought to perform one or a greater amount of the accompanying targets.

Convey data about past exercises, current status or projections of the Future.

Signal important occasions, opportunities, issues, or notices.

Trigger an activity.

Confirm an action.

5. SOFTWARE ENVIRONMENT

5.1. JAVA TECHNOLOGY

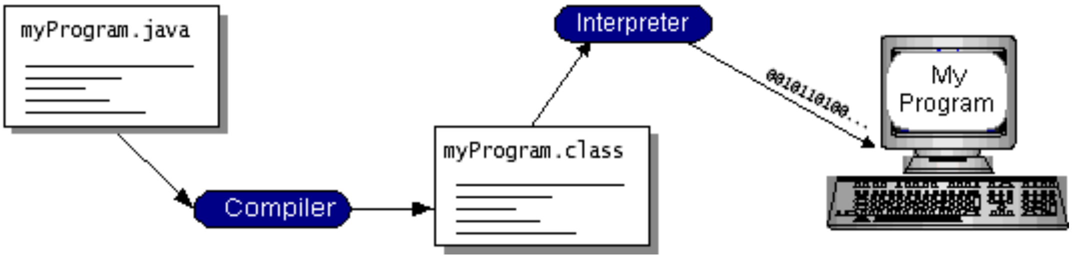
Java technology is both a programming language and a platform.

5.1.1. The Java Programming Language

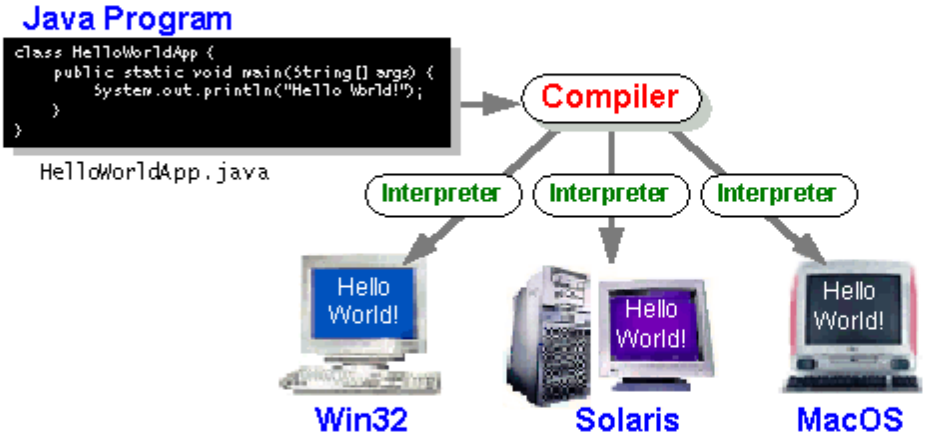
The Java programming language is a high-level language that can be characterized by all of the followings:

1. Simple
2. Architecture neutral
3. Object oriented
4. Portable
5. Distributed
6. High performance
7. Interpreted
8. Multithreaded
9. Robust
10. Dynamic
11. Secure

With most programming dialects, you either incorporate or decipher a system so you can run it on your PC. The Java programming dialect is uncommon in that a system is both arranged and translated [32]. With the compiler, first you make an interpretation of a system into a middle dialect called Java byte codes — the stage free codes translated by the translator on the Java stage[33]. The mediator parses and runs every Java byte code direction on the PC. Assemblage happens just once; understanding happens every time the project is executed. The accompanying figure outlines how this functions.



You can consider Java byte codes as the machine code guidelines for the Java Virtual Machine (Java VM). Each Java mediator, whether it's an advancement device or a Web program that can run applets, is an execution of the Java VM. Java byte codes make "compose once, run anyplace" conceivable[34]. You can aggregate your project into byte codes on any stage that has a Java compiler. The byte codes can then be keep running on any usage of the Java VM. That implies that the length of a PC has a Java VM, the same project written in the Java programming dialect can keep running on Windows 2000, a Solaris workstation, or on an iMac.



```
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

5.1.2. JAVA PLATFORM

A stage is the equipment or programming environment in which a project runs. We've as of now specified probably the most mainstream stages like Windows 2000, Linux, Solaris, and MacOS[35][36]. Most stages can be portrayed as a mix of the working framework and equipment. The Java stage varies from most different stages in that it's a product just stage that keeps running on top of other equipment based stages.

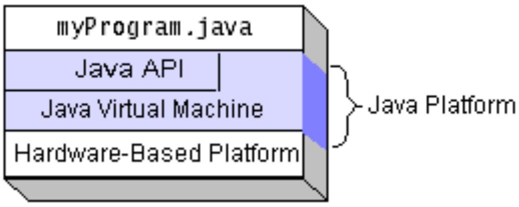
The Java platform has two components:

- *Java Virtual Machine* (Java VM)
- *Java Application Programming Interface* (Java API)

You've as of now been acquainted with the Java VM. It's the base for the Java stage and is ported onto different equipment based stages.

The Java API is a vast gathering of instant programming parts that give numerous valuable abilities, for example, graphical client interface (GUI) gadgets. The Java API is assembled into libraries of related classes and interfaces; these libraries are known as bundles.[37] The following segment, What Can Java Technology Do? Highlights what usefulness a portion of the bundles in the Java API give.

The accompanying figure delineates a system that is running on the Java stage. As the figure demonstrates, the Java API and the virtual machine protect the project from the equipment.



Native code will be code that after you accumulate it, the gathered code keeps running on a particular equipment stage. As a stage free environment, the Java stage can be a bit slower than local code. In any case, brilliant compilers, very much tuned translators, and in the nick of time byte code compilers can convey execution near that of local code without debilitating transportability[38].

What Can Java Technology Do?

The most widely recognized sorts of projects written in the Java programming dialect are applets and applications. On the off chance that you've surfed the Web, you're likely effectively acquainted with applets. An applet is a system that holds fast to specific traditions that permit it to keep running inside a Java-empowered program.

Not with standing, the Java programming dialect is not only to write adorable, captivating applets for the Web. The universally useful, abnormal state Java programming dialect is additionally a capable programming stage. Utilizing the liberal API, you can compose numerous sorts of projects.

An application is a standalone program that runs specifically on the Java stage. An exceptional sort of use known as a server serves and backings customer on a system. Cases of servers are Web servers, intermediary servers, mail servers, and print servers. Another specific project is a servlet[39][40]. A servlet can practically be considered as an applet that keeps running on the server side. Java Servlets are a mainstream decision for building intuitive web applications, supplanting the utilization of CGI scripts. Servlets are like applets in that they are runtime augmentations of utilizations. Rather than working in programs, however, servlets keep running inside Java Web servers, designing or customizing the server [41]

How does the API bolster every one of these sorts of projects? It does as such with bundles of programming segments that gives an extensive variety of usefulness. Each full usage of the Java stage gives you the accompanying elements:

The essentials: Objects, strings, strings, numbers, information and yield, information structures, framework properties, date and time, etc.

APPLETS: The arrangement of traditions utilized by applets.

Organizing: URLs, TCP (Transmission Control Protocol), UDP (User Data gram Protocol) attachments, and IP (Internet Protocol) addresses[42].

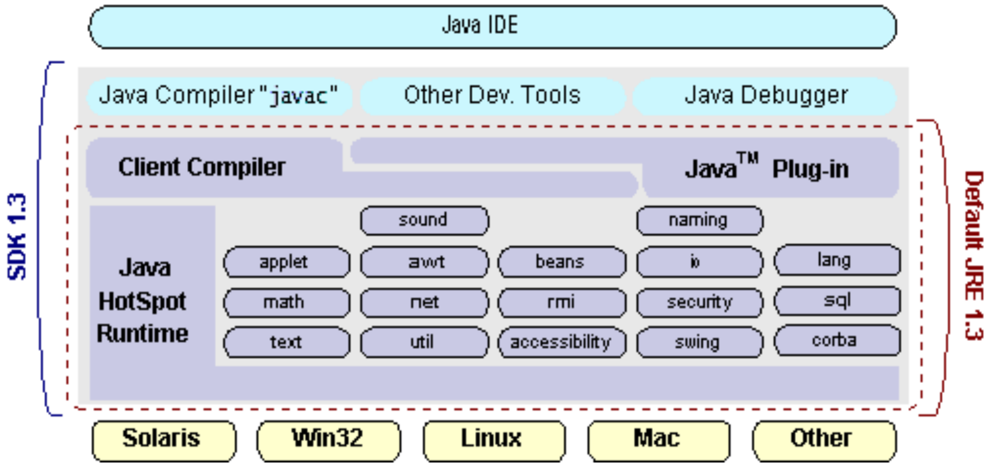
INTERNATIONALIZATION: Help for composing programs that can be confined for clients around the world. Projects can consequently adjust to particular districts and be shown in the suitable dialect.

SECURITY: Both low level and abnormal state, including electronic marks, open and private key administration, access control, and declarations.

Programming COMPONENTS: Known as JavaBeansTM, can connect to existing segment designs.

OBJECT SERIALIZATION: Allows lightweight industriousness and correspondence by means of Remote Method Invocation (RMI).

JAVA DATABASE CONNECTIVITY (JDBCTM): Provides uniform access to an extensive variety of social databases[43]. The Java stage likewise has APIs for 2D and 3D design, openness, servers, joint effort, telephony, discourse, liveliness, and that's only the tip of the iceberg. The accompanying figure delineates what is incorporated into the Java 2 SDK[44][45].



How Will Java Technology Change My Life?

We can't guarantee you notoriety, fortune, or even an occupation on the off chance that you take in the Java programming dialect. Still, it is liable to improve your projects and requires less exertion than different dialects. We trust that Java innovation will help you do the accompanying:

Begin rapidly: Although the Java programming dialect is an intense item situated dialect, it's anything but difficult to learn, particularly for software engineers effectively acquainted with C or C++ [46]

Compose less code: Comparisons of system measurements (class tallies, strategy numbers, et cetera) propose that a project written in the Java programming dialect can be four times littler than the same system in C++.

Compose better code: The Java programming dialect empowers great coding practices, and its waste accumulation helps you stay away from memory spills. Its item introduction, its JavaBeans segment design, and its colossal, effortlessly extendible API let you reuse other individuals' tried code and present less bugs[47].

Create programs all the more rapidly: Your improvement time might be as much as twice as quick versus composing the same project in C++. Why? You compose less lines of code and it is a less difficult programming dialect than C++.

Maintain a strategic distance from stage conditions with 100% Pure Java: You can keep your project convenient by staying away from the utilization of libraries written in different dialects[48]. The 100% Pure JavaTM Product Certification Program has a vault of authentic procedure manuals, white papers, leaflets, and comparative materials on the web.

Compose once, run anyplace: Because 100% Pure Java projects are arranged into machine-autonomous byte codes, they run reliably on any Java stage.

Convey programming all the more effectively: You can update applets effortlessly from a focal server. Applets exploit the component of permitting new classes to be stacked "on the fly," without recompiling the whole program[49].

Furthermore, for powerfully overhauling the store table we go for MS Access database.

Java have two things: a programming language and a platform.

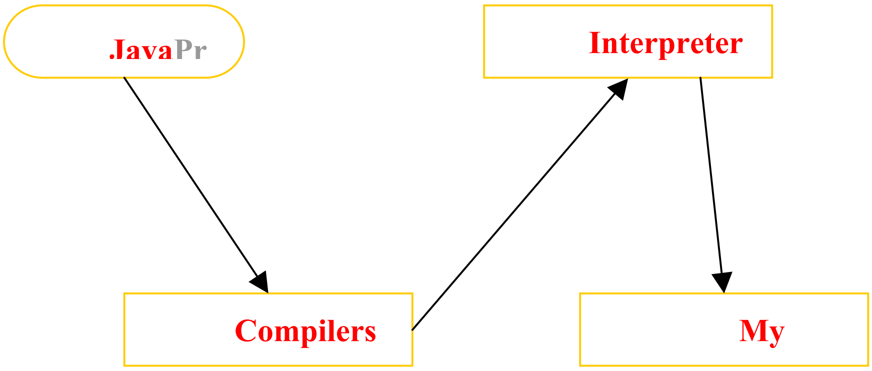
Java is a high-level programming language that is all of the following :

- Simple Architecture-neutral
- Object-oriented Portable
- Distributed High-performance

- Interpreted multithreaded
- RobustDynamic
- Secure

Java is additionally irregular in that every Java project is both ordered and deciphered. With an incorporate you make an interpretation of a Java program into a middle of the road dialect called Java byte codes the stage free code guideline is passed and keep running on the PC.

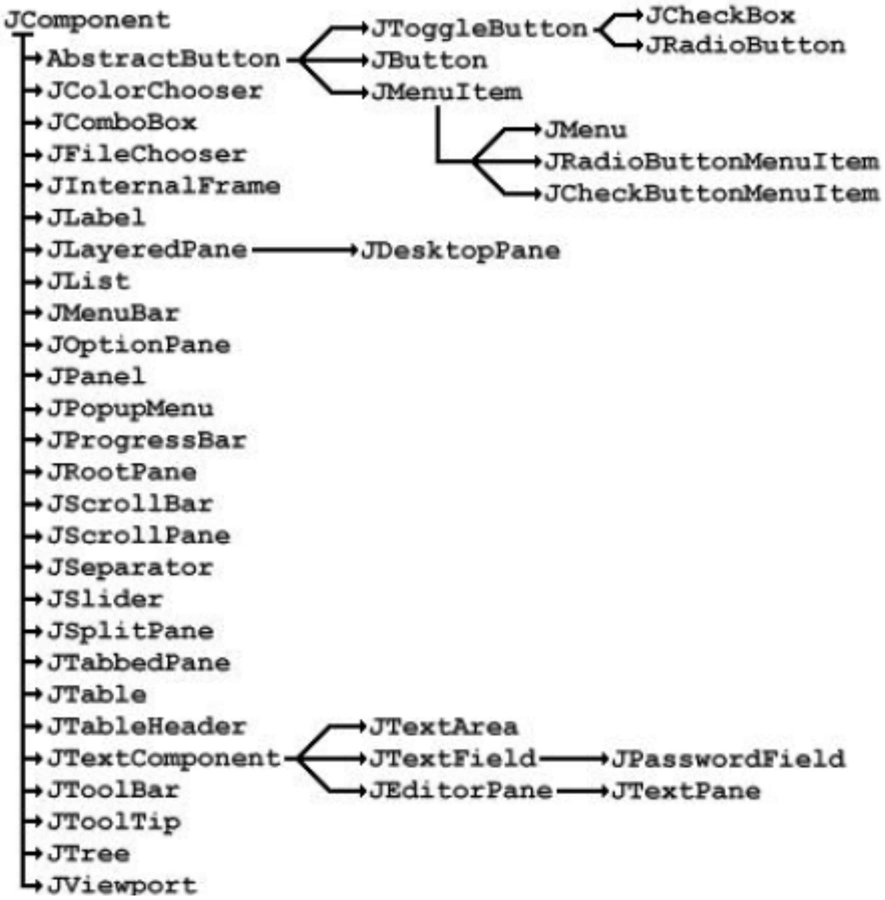
Aggregation happens just once; understanding happens every time the system is executed. The figure outlines how this functions[67].Java byte codes make "compose once, run anyplace" conceivable.



You can assemble your Java program into byte codes on my stage that has a Java compiler. The byte codes can then be run any execution of the Java VM. For instance, the same Java project can run Windows NT, Solaris, and Macintosh[68].

5.2. SWING

Swing is an extensive arrangement of parts going from the exceptionally basic, for example, names, to the extremely mind boggling, for example, tables, trees, and styled content reports. All Swing parts are gotten from a solitary guardian called JComponent which broadens the AWT Container class. In this manner, Swing is best portrayed as a layer on top of AWT instead of a swap for it[69]. Figure 1.2 demonstrates a halfway JComponent pecking order. On the off chance that you contrast this and the AWT Component heirarchy of figure 1.1 you will see that for each AWT part there is a Swing proportional with prefix "J". The main exemption to this is the AWT Canvas class, for which JComponent, JLabel, or JPanel can be utilized as a substitution (in segment 2.8 we talk about this in point of interest). You will likewise see numerous Swing classes with no AWT partners[70]. Figure 1.2 speaks to just a little division of the Swing library, yet this portion comprises of the classes you will be managing most. Whatever is left of Swing exists to give broad backing and customization capacities for the parts these classescharacterize.



5.2.1. SWING PACKAGE OVERVIEW

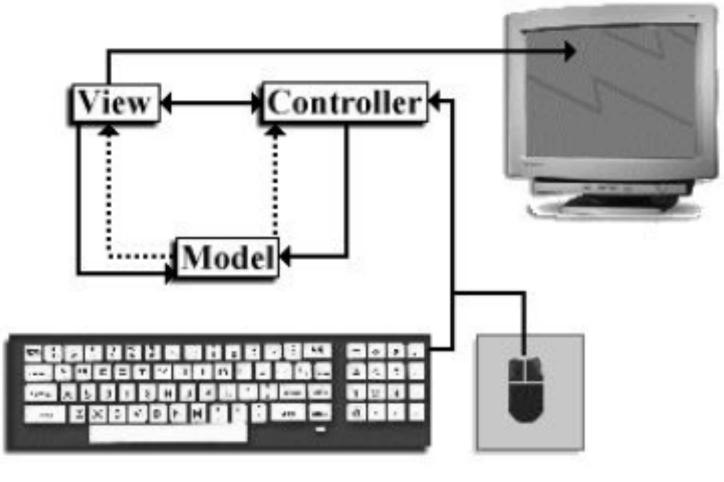
javax.swing Contains the most essential Swing segments, default part models, and interfaces[71]. (The greater part of the classes appeared in Figure 1.2 are contained in this bundle.) javax.swing.border Classes and interfaces used to characterize particular outskirt styles. Note that outskirts can be shared by any number of Swing segments, as they are not parts themselves. javax.swing.colorchooser Classes and interfaces supporting the JColorChooser segment, utilized for shading determination. (This bundle likewise contains some fascinating undocumented private classes.)

javax.swing.event The occasion bundle contains all Swing-particular occasion sorts and audience members. Swing segments likewise bolster occasions and audience members characterized in java.awt.event and java.beans. javax.swing.filechooser Classes and interfaces supporting the JFileChooser segment, utilized for record determination. javax.swing.plaf Contains the pluggable look-and-feel API used to characterize custom client interface segments. The vast majority of the classes in this bundle are conceptual. They are subclassed and executed by look-and-feel usage, for example, metal, theme, and fundamental. The classes in this bundle are proposed for use just by designers who, for some reason, can't expand on top of existing look-and-feels. javax.swing.plaf. fundamental Consists of the Basic look-and-feel usage which all look-and-feels furnished with Swing are based on top of. We are ordinarily anticipated that would subclass the classes in this bundle in the event that we need to make our own tweaked look-and-feel. javax.swing.plaf.metal Metal is the default look-and-feel of Swing parts. It is the main look-and-feel that boats with Swing not intended to be reliable with a particular stage. javax.swing.plaf.multi this is the Multiplexing look-and-feel. This is not a general look-and-feel usage in that it doesn't characterize the genuine look or feel of any segments. Or maybe, it gives the capacity to consolidate a few search and-feels for simultaneous use. A commonplace case may be utilizing a sound based look-and-feel in blend with metal or theme[72].

Right now Java 2 does not dispatch with any multiplexing look-and-feel implementations (notwithstanding, talk has it that the Swing group is chipping away at a sound look-and-feel as we think of this). javax.swing.table Classes and interfaces supporting the JTable control. This part is utilized to oversee even information in spreadsheet structure. It underpins a high level of customization without requiring look - and-feel upgrades. javax.swing.text Classes and interfaces utilized by the content segments including support for plain and styled records, the perspectives of those archives, highlighting, caret control and customization, proofreader activities and console customization. javax.swing.text.html This augmentation of the content bundle contains support for HTML content parts. (HTML backing was by and large totally revamped and developed while we were composing this book. As a result of this our scope of it is regrettably constrained.) javax.swing.text.html. parser Support for parsing HTML. javax.swing.text.rtf Contains support for RTF archives. javax.swing.tree Classes and interfaces supporting the JTree part. This part is utilized for the showcase and administration of hierarchical information. It bolsters a high level of customization without requiring look-and-feel upgrades. javax.swing.undo The fix bundle contains support for executing and overseeing fix/re-try usefulness.

5.3. MVC architecture

MVC is an understood item situated client interface outline decomposition that goes back to the late 1970s. Segments are separated into three sections: a model, a perspective, and a controller[73]. Every Swing part depends on a more advanced variant of this configuration. Before we examine how MVC functions in Swing, we have to see how it was initially intended to function.



MODEL

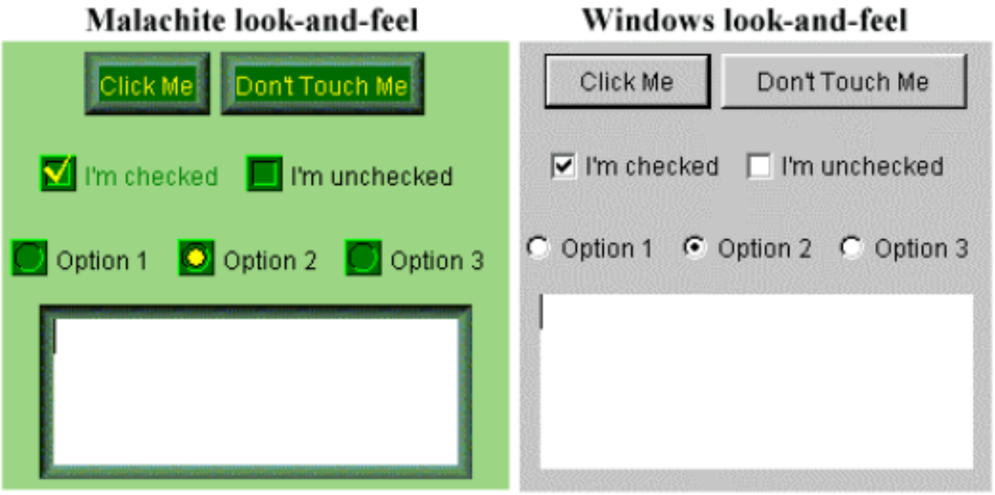
The model is in charge of keeping up all parts of the segment state. This incorporates, for instance, such values as the squeezed/unpressed condition of a push catch, a content part's character information and data about how it is organized, and so on[74]. A model might be in charge of aberrant correspondence with the perspective and the controller. By aberrant we imply that the model does not "know" its perspective and controller- - it doesn't keep up or retrieve references to them. Rather the model will convey warnings or shows (what we know as occasions). In figure 1.3 this aberrant correspondence is spoken to by dashed lines. View The perspective decides the visual representation of the segment's model. This is a part's "look." For instance, the perspective shows the right shade of a segment, whether the segment seems raised or brought down (on account of a catch), and the rendering of a coveted textual style. The perspective is in charge of keeping its on-screen representation overhauled and may do as such after getting aberrant messages from the model, or direct messages from the controller[75].

CONTROLLER

The controller is in charge of figuring out if the segment ought to respond to any info occasions from data gadgets, for example, the console or mouse[76]. The controller is the "vibe" of the segment, and it figures out what activities are performed when the part is utilized. The controller can get direct messages from the perspective, and roundabout messages from the model. For instance, assume we have a checked (chose) checkbox in our interface. On the off chance that the controller discovers that the client has performed a mouse click it might make an impression on the perspective. On the off chance that the perspective verifies that the snap happened on the checkbox it makes an impression on the model. The model then redesigns itself and telecasts a message, which will be gotten by the view(s), to let it know that it ought to upgrade itself taking into account the new condition of the model[77]. Along these lines, a model is not bound to a particular perspective or controller, permitting us to have a few perspectives and controller's controlling a solitary model.

CUSTOM VIEW AND CONTROLLER

One of the significant points of interest MVC engineering gives is the capacity to alter the "look" and "feel"of a segment without changing the model. Figure 1.4 demonstrates a gathering of segments utilizing two distinctive client interfaces[78]. The vital point to make about this figure is that the parts indicated are really the same; however they are demonstrated utilizing two distinctive look-and-feel executions (changed perspectives and controllers - talked about beneath).



Some Swing segments additionally give the capacity to redo particular parts of a segment without influencing the model. All the more particularly, these segments permit us to characterize custom cell renderers and editors used to show and acknowledge particular information individually[79]. Figure 1.5 demonstrates the sections of a table containing securities exchange information rendered with custom symbols and hues. We will look at how to exploit this usefulness in our investigation of Swing combo boxes, records, tables, and trees.

SWING MECHANICS

JComponent properties, sizing, and positioning • Event handling and dispatching

- Multithreading
- Timers
- AppContext & service classes
- Inside Timers & the TimerQueue
- JavaBeans
- Fonts, Colors, Graphics & text
- Using the Graphics clipping area

- Graphics Debugging
- Painting and Validation
- Focus Management
- Keyboard input, KeyStrokes, and Actions
- SwingUtilities.

JComponent properties, size, and positioning Properties

All Swing parts fit in with the JavaBeans particular[80]. In area we will talk about this in subtle element. Among the five components a JavaBean is relied upon to backing is an arrangement of properties and related accessor Methods. A property is a worldwide variable, and its accessor strategies, assuming any, are typically of the structure setPropertyname(), getPropertyname() or isPropertyname(). A property that has no occasion terminating connected with an adjustment in its quality is known as a straightforward property. A bound property is one for which PropertyChangeEvents are let go after it changes state. We can enroll PropertyChangeListeners to listen for PropertyChangeEvents through JComponent's addPropertyChangeListener() technique. An obliged property is one for which PropertyChangeEvents are terminated before an adjustment in state happens. We can enlist VetoableChangeListeners to listen for PropertyChangeEvents through JComponent's addVetoableChangeListener() strategy. A change can be vetoed in the occasion taking care of code of a VetoableChangeListener by tossing a PropertyVetoException. (There is one and only Swing class with compelled properties: JInternalFrame). Note: Each of these occasion and audience classes is characterized in the java.awt.beans bundle. PropertyChangeEvent's convey three bits of data with them: name of the property, old quality, and new esteem. Beans can utilize an example of PropertyChangeSupport to deal with the dispatching of PropertyChangeEvents comparing to every bound property, to each enrolled audience. Essentially, an occurrence of VetoableChangeSupport can be utilized to deal with the sending of all PropertyChangeEvents comparing to each compelled property.

Swing presents another class called SwingPropertyChangeSupport (characterized in javax.swing.event) which is a subclass of, and verging on indistinguishable to, PropertyChangeSupport. The distinction is that SwingPropertyChangeSupport has been worked to be more productive. It does this by yielding string wellbeing, which, as we will see later in this part, is not an issue in Swing if the multithreading rules are taken after reliably (in light of the fact that all occasion preparing ought to happen on one and only string - the occasion dispatching string). So in the event that we are certain that our code has been built in a string safe manner, we are urged to utilize this more efficient adaptation, as opposed to PropertyChangeSupport. Note: There is no Swing likeness VetoableChangeSupport in light of the fact that there are as of now just four obliged properties in Swing- - all characterized in JInternalFrame[81]

Swing additionally presents another sort of property which we will call a change property, for absence of a given name. We utilize Change Listeners to listen for Change Events that get let go when these properties change state. A Change Event just conveys one bit of data with it: the wellspring of the occasion. Hence, change properties are less effective than bound or obliged properties, yet they are more across the board. A JButton, for example, sends change occasions at whatever point it is outfitted (squeezed surprisingly), squeezed, and discharged (see part 5). Another new property-like component Swing presents is the idea of customer properties[82]. These are fundamentally key/esteem sets put away in a Hashtable gave by every Swing part. This permits properties to be included and evacuated at run-time, and is regularly a helpful spot to store information without building a different subclass.Warning: Client properties may appear like an awesome approach to include property change support for custom segments, however we are unequivocally exhorted against this: "The clientProperty word reference is not expected to bolster huge scale augmentations to JComponent nor if it be viewed as a distinct option for subclassing when planning another component[83]."API Client properties are bound properties: when a customer property changes, a PropertyChangeEvent is dispatched to all enrolled PropertyChangeListeners. To add a property to a part's customer properties Hashtable, we do the accompanying: myComponent.putClientProperty("myname", myValue);

5.4. SYSTEM TESTING

The motivation behind testing is to find blunders. Testing is the procedure of attempting to find each possible shortcoming or shortcoming in a work item. It gives an approach to check the usefulness of parts, sub gatherings, congregations and/or a completed item. It is the procedure of practicing programming with the purpose of guaranteeing that the

Programming framework lives up to its prerequisites and client desires and does not fizzle in an unsatisfactory way. There are different sorts of test. Every test sort addresses a particular testing prerequisite[88][89].

UNIT TESTING

Unit testing includes the configuration of experiments that accept that the inward program rationale is working appropriately, and that program inputs produce substantial yields. All choice branches and inward code stream ought to be approved. It is the trying of individual programming units of the application .it is done after the finish of an individual unit before joining[90]. This is a basic testing, that depends on learning of its development and is intrusive. Unit tests perform fundamental tests at segment level and test a particular business procedure, application, and/or framework arrangement. Unit tests guarantee that every one of a kind way of a business process performs precisely to the archived determinations and contains unmistakably characterized inputs and expected results.

Integration testing

Integration tests are intended to test incorporated programming parts to figure out whether they really keep running as one project[91]. Testing is occasion driven and is more worried with the fundamental result of screens or fields. Coordination tests exhibit that in spite of the fact that the parts were independently fulfillment, as appeared by effectively unit testing, the blend of segments is right and predictable. Incorporation testing is particularly gone for uncovering the issues that emerge from the blend of segments.

Functional test

Functional tests give methodical exhibits that capacities tried are accessible as determined by the business and specialized prerequisites, framework documentation, and client manuals[92].

Functional testing is focused on the accompanying things:

- ***Valid Input*** : recognized classes of legitimate data must be acknowledged.
- ***Invalid Input*** : recognized classes of invalid info must be rejected.
- ***Functions***: distinguished capacities must be worked out.
- ***Output*** : distinguished classes of use yields must be worked out. Frameworks/Procedures: interfacing frameworks or strategies must be summoned.

Association and arrangement of practical tests is centered around necessities, key capacities, or unique experiments. What's more, efficient scope relating to recognize Business process streams; information fields, predefined forms, and progressive procedures must be considered for testing. Before useful testing is finished, extra tests are recognized and the successful estimation of current tests is resolved.

SYSTEM TESTING

System testing guarantees that the whole incorporated programming framework meets necessities[93]. It tests a design to guarantee known and unsurprising results. A case of framework testing is the setup arranged framework mix test. Framework testing depends on procedure portrayals and streams, stressing pre-driven procedure connections and combination focuses.

WHITE BOX TESTING

White Box Testing is a trying in which in which the product analyzer knows about the internal workings, structure and dialect of the product, or possibly its motivation. It is reason. It is utilized to test zones that can't be come to from a discovery level.

DISCOVERY TESTING

Discovery Testing will be trying the product with no learning of the internal workings, structure or dialect of the module being tried. Discovery tests, as most different sorts of tests, must be composed from a conclusive source record, for example, determination or necessities report, for example, particular or prerequisites archive. It is a trying in which the product under test is dealt with, as a discovery .you can't "see" into it[94]. The test gives inputs and reacts to yields without considering how the product functions.

UNIT TESTING:

Unit testing is normally led as a major aspect of a consolidated code and unit test period of the product lifecycle, in spite of the fact that it is not extraordinary for coding and unit testing to be led as two unmistakable stages[95].

TEST STRATEGY AND APPROACH

Field testing will be performed physically and practical tests will be composed in point of interest.

Test goals

- All field sections must work appropriately.
- Pages must be enacted from the recognized connection.
- The passage screen, messages and reactions must not be deferred.
- Components to be tried
- Verify that the passages are of the right configuration
- No copy passages ought to be permitted
- All connections ought to take the client to the right page.

Integration Testing

Programming joining testing is the incremental coordination testing of two or more incorporated programming segments on a solitary stage to deliver disappointments brought about by interface deformities[96]. The undertaking of the coordination test is to watch that parts or programming applications, e.g. parts in a product framework or – one stage up – programming applications at the organization level – interface without mistake.

Test Results: All the experiments said above passed effectively. No imperfections experienced.

Acceptance Testing

Client Acceptance Testing is a basic period of any venture and requires noteworthy interest by the end client. It additionally guarantees that the framework meets the utilitarian necessities[97].

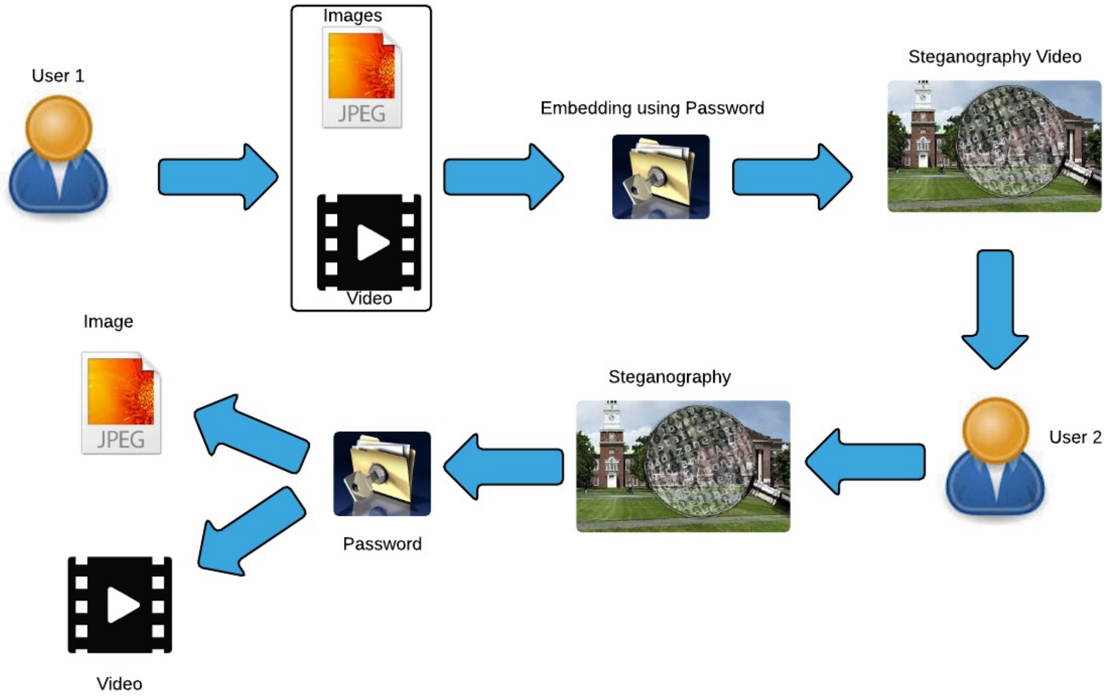
Test Results: All the experiments said above passed effectively. No deformities experienced.

TEST CASE REPORTS:

Object name	**Test ID**	**Test case description**	**Action**	**Expected result**	**Actual result**	**Status**
Secret Image	TC 001	To check whether the given Secret Image is empty or not	Click on Submit or reset button	Should intimate to fill the field	Same as expected	Pass
Video file	TC 002	To check whether the given video file is empty or not	Click on Submit or reset button	Should intimate to fill the field	Same as expected	Pass
Destination Path	TC 003	To check whether the given Dest Path file	Click on Submit or reset button	Should intimate to fill the field are correct	Same as expected	Pass

6. Experiment

6.1.SYSTEM ARCHITECTURE:



MODULES

A. Secret Message Formulation

Secret Message Formulation is out a secret message which is a picture. Pixel estimations of first 8x8 of 128x128 measured picture are taken in. Every pixel force is then changed over into identical twofold values. As the measure of the picture is 128x128 we got 128x128x8=131072 piece (the secret message bits to be covered up) [19].

B. Frame Extraction and Embedding Secret Message

Here we have taken an Avi video record as a spread or host video and all casings are extricated (28 outlines) [20]. The determination of the first AVI is 120x160 pixels. The R-channel is utilised for encoding a secret message in the wake of performing piece DCT on those casings. As the span of unique babra.bmp picture is 128x128, thus we need to encode all out 128x128x8 bits in the video outlines [21]. Here we insert 16 bits for every 8x8 DCT higher request coefficient and in a special casing we can install edges can suit our secret message bits. After removing the casings, every R-channel casing is square prepared by 8x8 DCT and 16-bit secret message bits are installed into the higher request DC co-productive of every piece. In the wake of encoding the R-channels of edges, we consolidate those to get the video AVI record with mystery message implanted. Figure 8 demonstrates the video stream outlines after secret message inserted. We didn't discover a great deal more bending in the video[22].

C. Decoding and Reconstruction of Secret Message

Decoding is done backwards method for encoding.

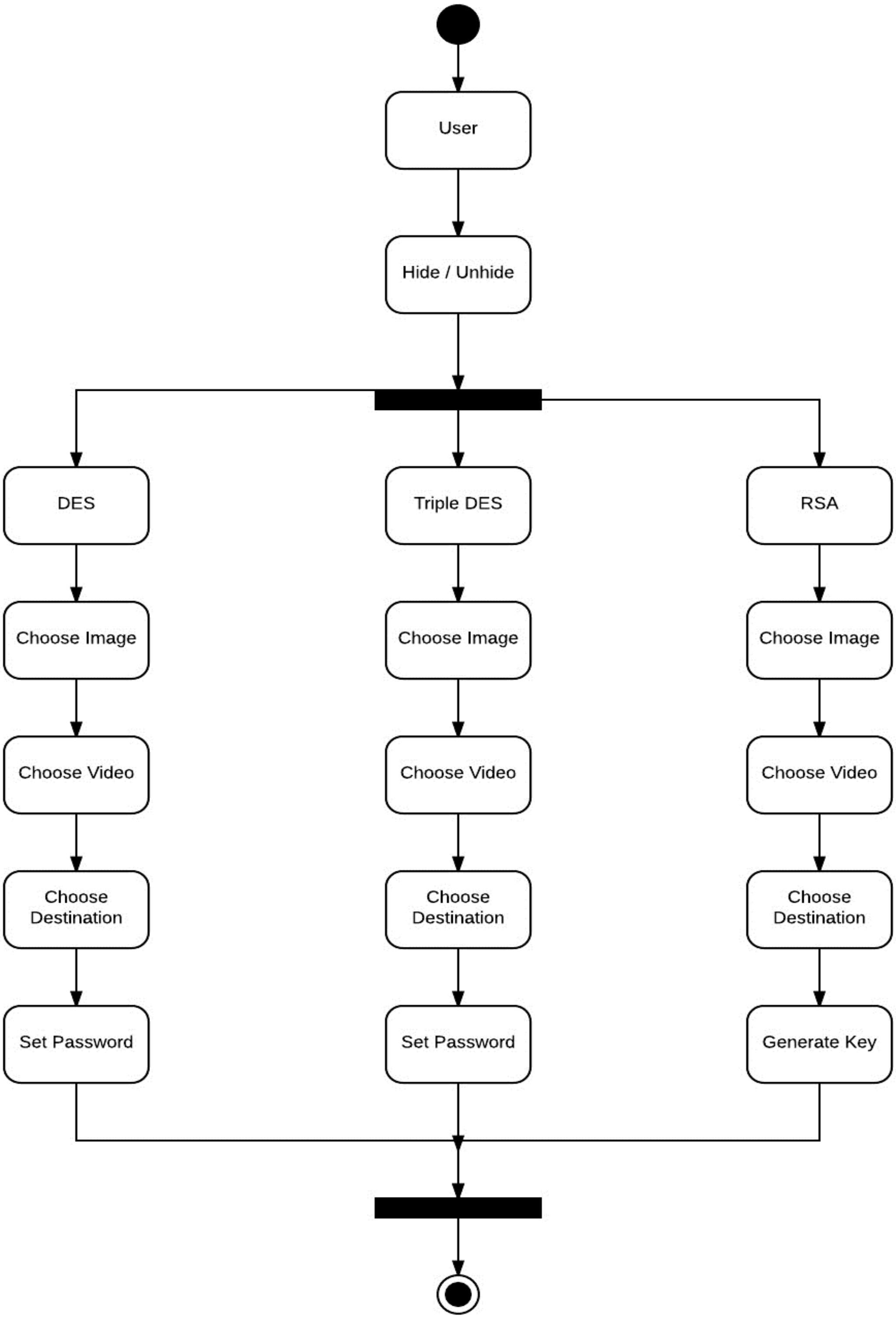
Step 1: First video edges are extricated.

Step 2: R-channel edges are handled by 8x8 piece DCT

Step 3: 8x8 square handled R-Channel unique casing qualities are subtracted to get the secret message.

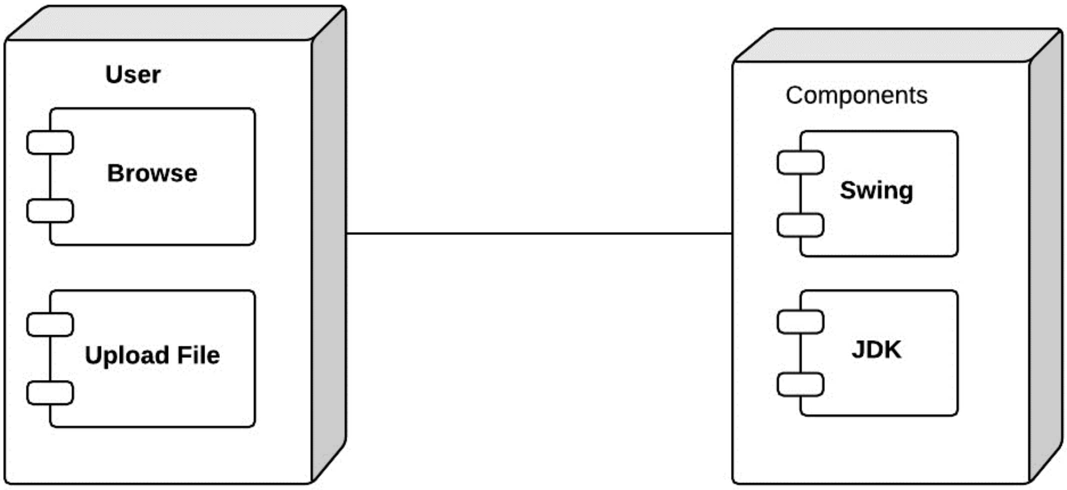
Step 4: From separated secret message the picture is recreated.

6.2. Flow chart

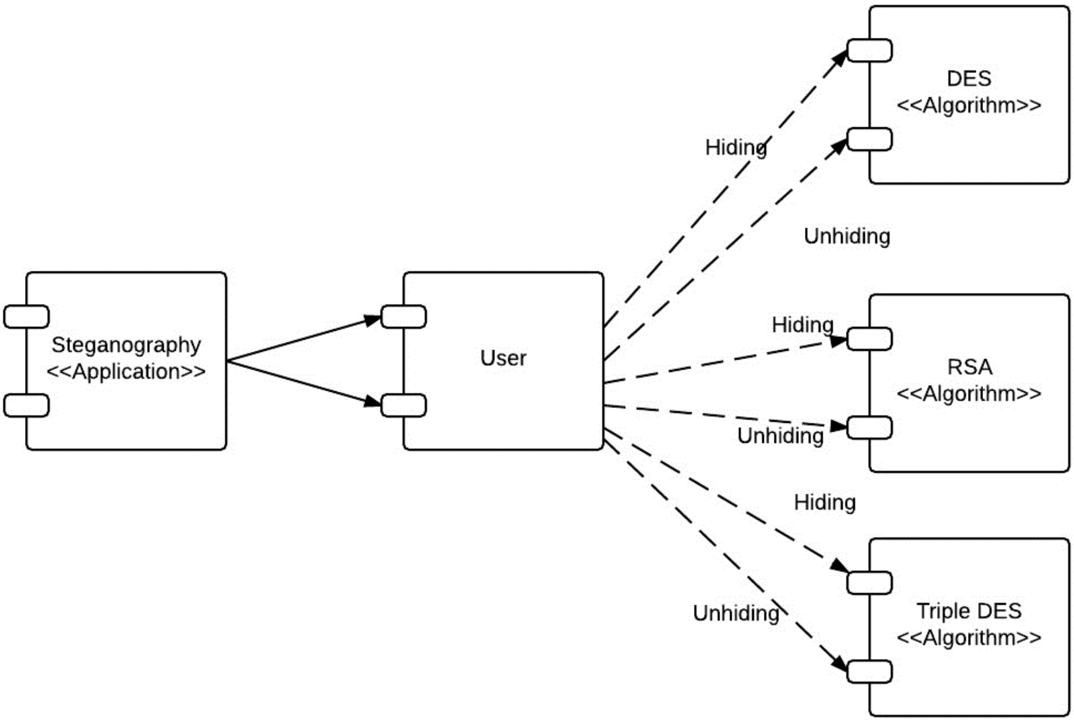


6.3. ACTIVITY DIAGRAM

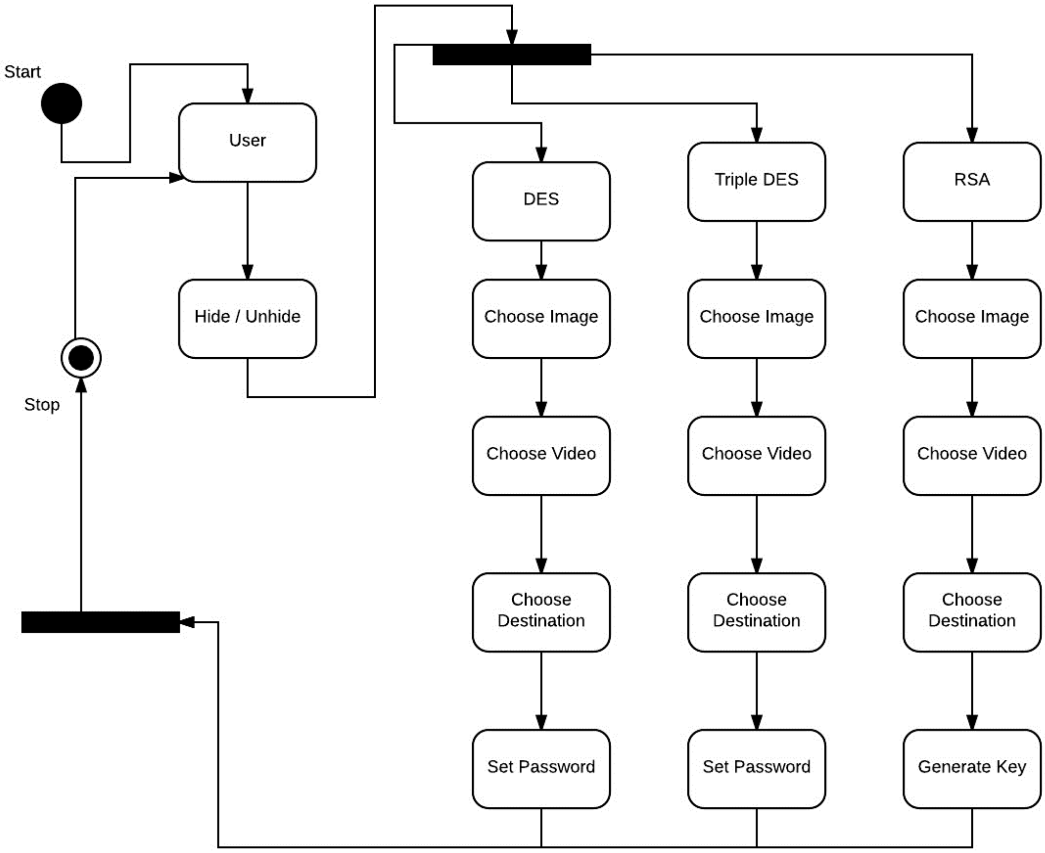
6.3.1. DEPLOYMENT DIAGRAM



6.3.2. COMPONENT DIAGRAM

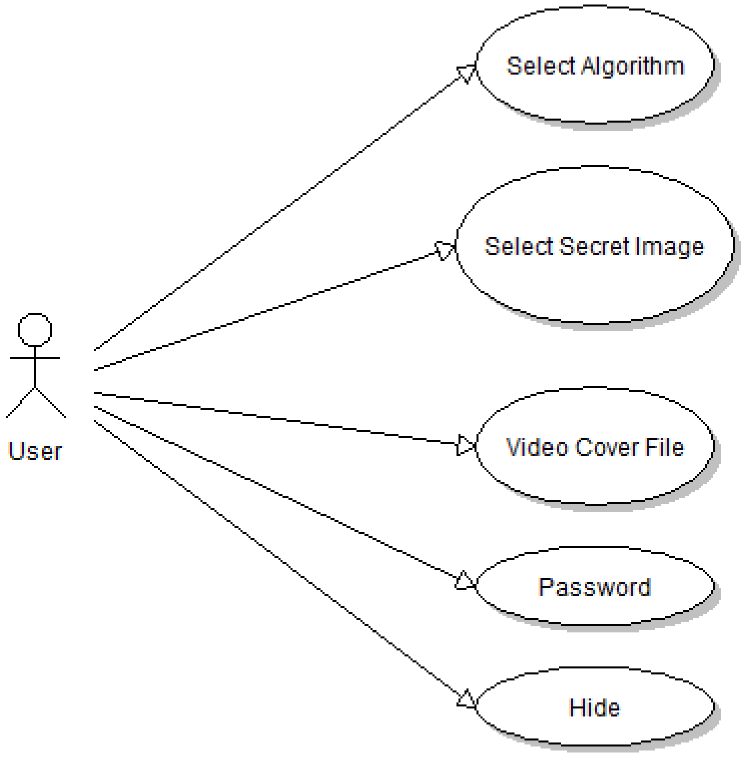


6.3.3. STATE DIAGRAM

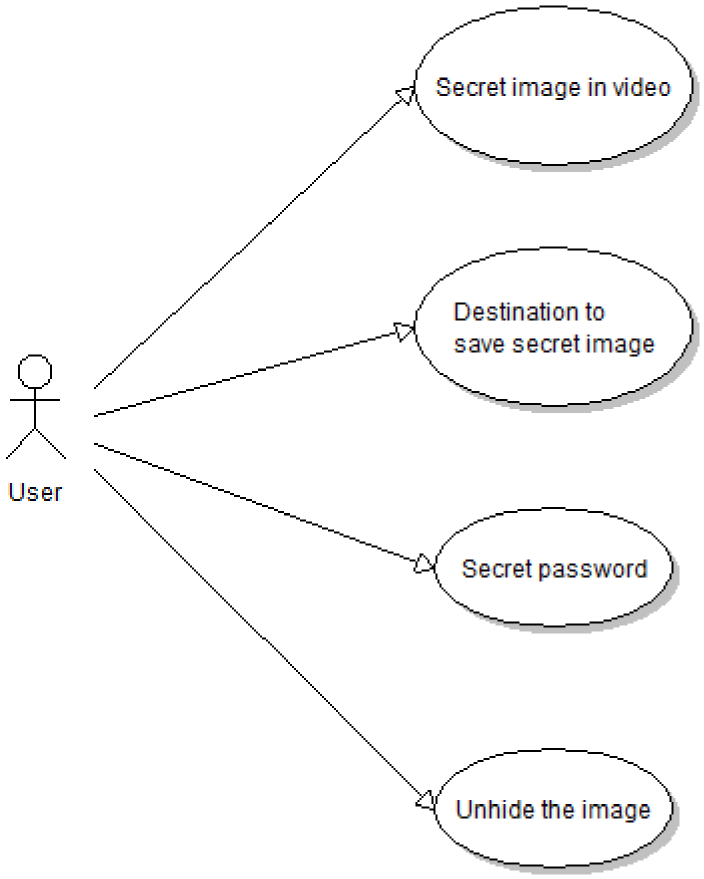


6.4. USECASE DIAGRAMS:

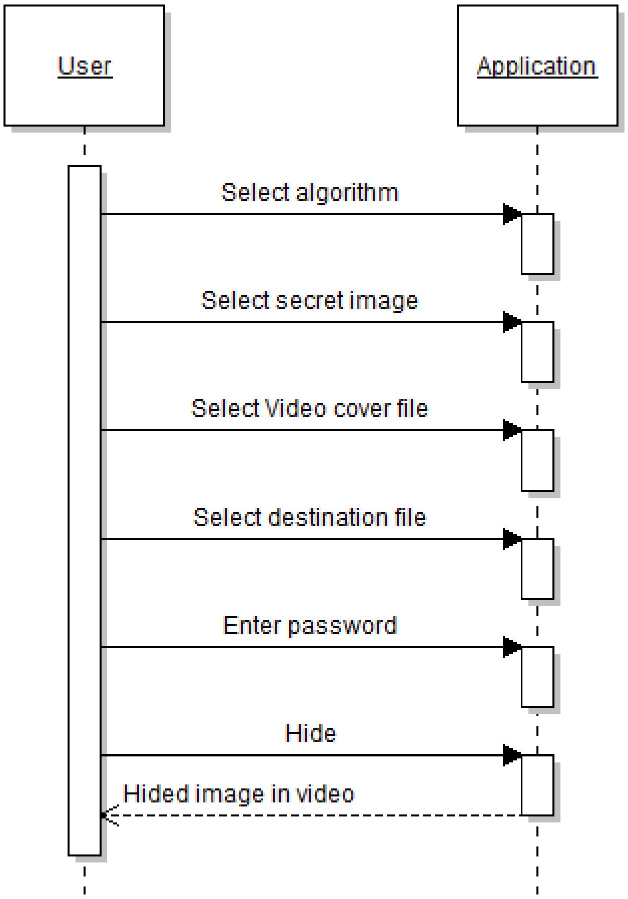
6.4.1. Level 1: HIDE SECRET IMAGE IN VIDEO



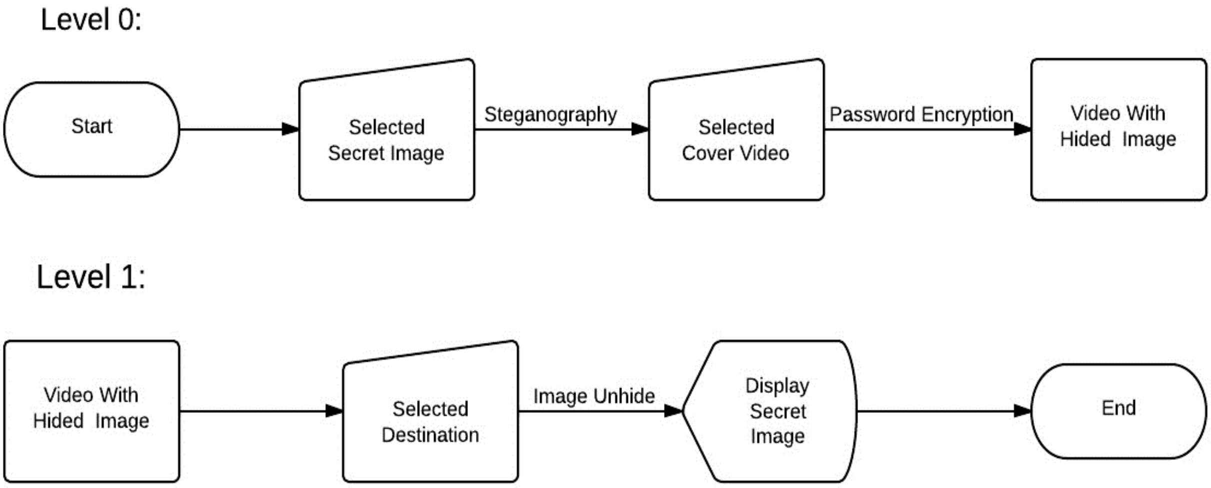
6.4.2. LEVEL 2: UNHIDE SECRET IMAGE FROM VIDEO

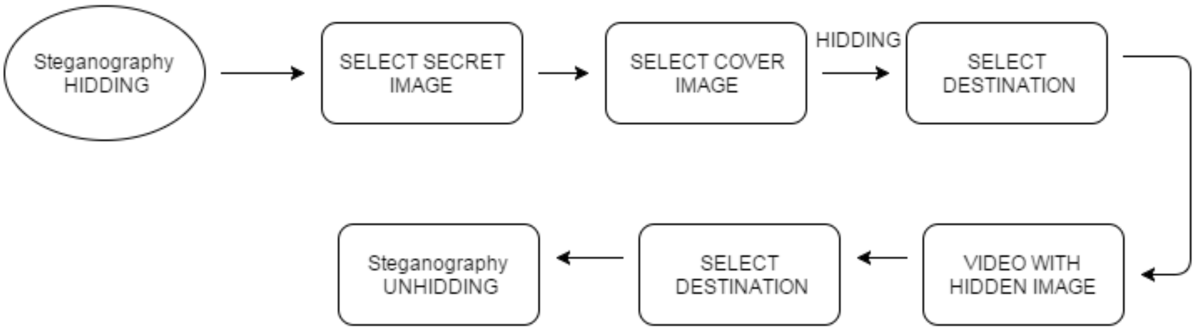


6.5. SEQUENCE DIAGRAM:



6.6. DATA FLOW DIAGRAM:




Steganography HIDDING
SELECT SECRET IMAGE
SELECT COVER IMAGE
HIDDING
SELECT DESTINATION
VIDEO WITH HIDDEN IMAGE
SELECT DESTINATION
Steganography UNHIDDING

7. Results

We use several mechanisms to determine the image quality of the stego synthetic texture. We define the first measurement, which is called the mean squared error of the overlapped area (MSEO) to determine the image quality of the synthetic results. MSEO reflects the similarity between the candidate patch and the synthesized area where we will specifically operate the image quilting technique during the message-oriented texture synthesis process. Consequently, the MSEO has a non-zero value even in the case of the pure patch based texture synthesis. If the MSEO produces a small value, it implies that the synthetic texture shows a high image quality of the overlapped areas. Obviously, the lower the MSEO value, the higher quality of the synthetic texture image. The equation of MSEO is shown in (6), where OL_istands for the overlapped area of the working location, i, p_{cj} stands for the pixel j of the candidate patch in OL_i, and p_{sj}stands for the pixel j of the synthesized area in OL_i.

$$\mathrm{MSEO} = \frac{1}{(EPn) \times Pw \times Ph} \sum_{i=1}^{EPn} \sum j \epsilon OLi \, (Pcj - psj\,)^2$$

The MSEO can be calculated during the texture synthesis procedure. When we synthesize a patch, we accumulate the squared errors of all the pixels in the overlapped area between the synthesized area and the selected candidate patch. When the texture synthesis procedure is finished, we can divide the summation of squared errors by the patch size and the number of the synthesized patches to produce the MSEO. It comes as no surprise that the MSEO is dependent on the capacity per patch. Table compares the MSEO to the embedding capacity for the four test source textures. When we consider the same patch size, the MSEO increases as we convey more secret bits per patch. These MSEO values coincide with the image quality when visualizing the stego synthetic texture. For a larger patch size, the boundary depth increases leading to a larger MSEO. Taking the test source texture “wheat grain” as an example, the MSEO is 1034.6 when conveying 5 bits of secret message perpatch, while it shows the MSEO of 1236.9 when conveying10 bits per patch.

	Patch Size $P_w*P_{h=}24*24$, Boundary Depth: P_d=4			Patch Size $P_w*P_{h=}48*48$, Boundary Depth: P_d=8		
	Pure	5BPP	10BPP	Pure	5BPP	10BPP
Mesh net	651.2	899.4	1327.6	936.3	1224.2	1565.3
Wood	595.6	795.2	1096.4	837.4	1132.3	1391.1
Wheat grain	547.8	776.9	1053.1	918.2	1034.6	1236.9

8. CONCLUSION:

In this Master thesis I have achieved a brilliant analysis and procurement of Steganographical concepts criteria, and further more by surface amalgamation and texture synthesis I have concluded these two important aspects

Analysis of the existing approaches available to perform stegnography has been performed to solid understanding and knowledge grasp that helps to develop a new strategy.

This paper proposes reversible steganographic algorithm using surface amalgamation and texture synthesis given an original source texture, this scheme can produce a large stego synthetic texture concealing secret messages.

REFERENCES

[1] N.F. Johnson and S. Jajodia, Exploring Steganography: Seeing the unseen, IEEE Computer, 41(2)(1998) 25- 44.

[2] J.C. Judge, Steganography: Past, present, future. SANS Institute publication, http://www.sans.org/reading_room/whitepapers/stenganography/552.php, 2001.

[3] N. Provos and P. Honeyman, Hide and seek: An introduction to Steganography, IEEE Security and Privacy, 01 (4)(2004)42-44.

[4] P. Moulin and R. Koetter, Data-hiding codes, Proceedings of the IEEE, 94 (12)(2005)2084-2125.

[5] S.B. Sadkhan, Cryptography: Current status and future trends, in: Proceedings of IEEE International Conference on Information & Communication Technologies: From Theory to Applications, Damascus. Syria, April 19-24, 2004, pp. 417-418.

[6] S. Lyu and H. Farid, steganalysis using higher-order image statistics, IEEE Transactions on Information Forensics and Security, 1(1)(2005) 111-119.

[7] D. Kahn, The Codebreakers: The comprehensive history of secret communication from ancient times to the Internet, Scribner, December 5, 1995.

[8] J.P. Delahaye, 'Information noyée, information cache', Pour la Science, 229(1995)142-145. www.apprendre-en-ligne.net/crypto/stegano/229_142_145.pdf. [in French].

[9] G.J. Simmons, The prisoners' problem and the subliminal channel, in: Proceedings of International conference on Advances in Cryptology, CRYPTO84, August 22-24, 1984, pp. 51-57.

[10] C. Kurak and J. McHugh, A cautionary note on image downgrading, in: Proceedings of the IEEE 8th Annual Computer Security Applications Conference, 30 Nov-4 Dec, 1992, pp. 154-159.

[11] T.L. Thomas, Al Qaeda and the Internet: The danger of "cyberplanning", Parameters, US Army War College Quarterly-Spring 2004. Available from: www.carlisle.army.mil/usawc/Parameters/04spring/thomas.pdf.

[12] C. Hosmer, Discovering hidden evidence, Journal of Digital Forensic Practice, (1)(2005)47-55.

[13] J.C. Hernandez-Castro, I. Blasco-Lopez and J.M. Estevez-Tapiador, Steganography in games: A general methodology and its application to the game of Go, Computers and Security, Elsevier Science, 25(2005) 54- 71.

[14] P. Hayati, V. Potdar, E. Chang, A survey of Steganography and steganalytic tools for the digital forensic investigator, available from: http://debii.curtin.edu.au/~pedram/images/docs/survey_of_steganography_and_steganalytic_tools.pdf

[15] W. Bender, W. Butera, D. Gruhl, R. Hwang, F.J. Paiz and S. Pogreb, Applications for data hiding, IBM Systems Journal, 49 (3&4)(2000) 547-558.

[16] F.A.P. Petitcolas, "Introduction to information hiding", in: S. Katzenbeisser and F.A.P. Petitcolas, (ed.) (2000) Information hiding techniques for Steganography and digital watermarking, Norwood: Artech House, INC.

[17] S. Miaou, C. Hsu, Y. Tsai and H. Chao, A secure data hiding technique with heterogeneous data-combining capability for electronic patient records, in: Proceedings of the IEEE 22nd Annual EMBS International Conference, July 24-28, 2000, Chicago, USA, pp. 280-284.

[18] U.C. Nirinjan and D. Anand, Watermarking medical images with patient information, in: Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Hong Kong, China, 29 Oct-1 Nov, 1998, pp. 704-705.

[19] Y. Li, C. Li and C. Wei, Protection of mammograms using blind Steganography and watermarking, in: Proceedings of the IEEE International Symposium on Information Assurance and Security, 2007, pp. 495- 499.

[20] D. Frith, Steganography approaches, options, and implications, Network Security, 2007(8)(2007)4-7.

[21] H. Farid, A Survey of image forgery detection, IEEE Signal Processing Magazine, 25(2)(2009)15-25.

[22] A. Cheddad, J. Condell, K. Curran and P. Mc Kevitt, A secure and improved self-embedding algorithm to combat digital document forgery, Signal Processing, 89 (12)(2009)2424-2442.

[23] N.F. Johnson and S.C. Katzenbeisser, "A survey of Steganography techniques", in: S. Katzenbeisser and F.A.P. Petitcolas, (ed.) (2000) Information hiding techniques for Steganography and digital watermarking, Norwood: Artech House, INC.

[24] K. Bailey and K. Curran, An evaluation of image based Steganography methods, Multimedia Tools and Applications, 40 (1)(2005) 55-88.

[25] [Hide and Seek]: ftp://ftp.funet.fi/pub/crypt/mirrors/idea.sec.dsi.unimi.it/cypherpunks/steganography/hdsk41b.zip [S-Tools]: ftp://ftp.funet.fi/pub/crypt/mirrors/idea.sec.dsi.unimi.it/code/s-tools4.zip [Stella] : http://wwwicg.informatik.uni-rostock.de/~sanction/stella/ [Hide in Picture]: http://sourceforge.net/projects/hide-in-picture/ [Revelation]: http://revelation.atspace.biz/ [Camouflage]: http://camouflage.unfiction.com/ [JpegX]: http://www.freewarefiles.com/Jpegx_program_19492.html [Data Stash]: http://www.skyjuicesoftware.com/software/ds_info.html [Other Tools]: http://www.jjtc.com/Security/stegtools.htm [F5]: http://wwwrn.inf.tu-dresden.de/~westfeld/f5.html [OutGuess]: http://www.outguess.org/

[26] P. Alvarez, Using extended file information (EXIF) file headers in digital evidence analysis, International Journal of Digital Evidence, Economic Crime Institute (ECI), 2(4)(2004)1-5.

[27] V.M. Potdar, S. Han and E. Chang, Fingerprinted secret sharing Steganography for robustness against image cropping attacks, in: Proceedings of IEEE 3rd International Conference on Industrial Informatics (INDIN), Perth, Australia, 10-12 August 2005, pp. 717-724.

[28] M.H. Shirali-Shahreza and M. Shirali-Shahreza, A new approach to Persian/Arabic text Steganography, in: Proceedings of 5th IEEE/ACIS International Conference on Computer and Information Science (ICISCOMSAR 2005), 10-12 July 2005, pp. 410-415.

[29] E.T. Lin and E.J. Delp, A review of data hiding in digital images, in: Proceedings of the Image Processing, Image Quality, Image Capture Systems Conference, PICS'99, the Society for Imaging Science and Technology, 1999, pp. 274-278.

[30] C.C. Chang, C.Y. Lin and Y.Z. Wang, New image Steganography methods using run-length Approach, Information Sciences, 175 (22)(2005)

[31] R.J. Hwang, K.T. Shih, C.H. Kao and T.M. Chang, Lossy compression tolerant Steganography, in: Proceedings of the 1st International Conference on The Human Society and the Internet-Internet Related Socio-Economic Issues, Lecture Notes In Computer Science, 2001, vol. 2105, pp. 427-445.

[32] X. Kong, Z. Wang and X. You, steganalysis of palette images: Attack optimal parity assignment algorithm, in: Proceedings of 5th IEEE International Conference on Information, Communications and Signal Processing, 05-09 Dec 2005, pp. 850-854.

[33] J. Fridrich, M. Goljan and D. Hogeg, steganalysis of JPEG images: Breaking the F5 algorithm, in: Proceedings of Information Hiding: 5th International Workshop, IH 2002 Noordwijkerhout, The Netherlands, LNCS, Springer, October 7-9, 2002, 2578/2004, pp. 410-424.

[34] K.H. Jung and K.Y. Yoo, Data hiding method using image interpolation, Computer Standards and Interfaces, 41(2)(2009) 455-470.

[35] Z. Li, X. Chen, X. Pan and X. Zeng, Lossless data hiding scheme based on adjacent pixel difference, in: Proceedings of the International Conference on Computer Engineering and Technology, 2009, pp. 588-592.

[36] P. Tsai, Y.C. Hu and H.L. Yeh, Reversible image hiding scheme using predictive coding and histogram shifting, Signal Processing, 89(5)(2009)1129-1144.

[37] G. Cancelli, G. J. Doërr, M. Barni and I.J. Cox, A comparative study of r1 steganalyzers, in Proceedings of IEEE 10th Workshop on Multimedia Signal Processing, MMSP'08, 8-10 Oct. 2008, pp.791-795.

[38] A.C. Popescu, Statistical tools for digital image forensics, Ph.D. Dissertation, Department of Computer Science, Dartmouth College, USA, 2005. Available from: http://www.cs.dartmouth.edu/~farid/publications/apthesis05.html, on 15-05-07 at 12:20.

[39] X. Li and J. Wang, A Steganography method based upon JPEG and particle swarm optimization algorithm, Information Sciences, 177(15)(2007)4099-41091.

[40] C.C. Chang, T.S. Chen and L.Z. Chung, A Steganography method based upon JPEG and quantization table modification, Information Sciences, 141 (1-2)(2002)124-148.

[41] A.M. Fard, M. Akbarzadeh-T and F. Varasteh-A, A new genetic algorithm approach for secure JPEG Steganography, in: Proceedings of IEEE International Conference on Engineering of Intelligent Systems, 22-24 April 2005, pp. 1-5.

[42] A.I. Hashad, A.S. Madani and A.E.M.A. Wahdan, A robust Steganography technique using discrete cosine transform insertion, in: Proceedings of IEEE/ITI 3rd International Conference on Information and Communications Technology, Enabling Technologies for the New Knowledge Society, 5-6 Dec. 2005, pp. 255-264.

[43] K.B. Raja, C.R. Chowdary, K.R. Venugopal and L.M. Patnaik, A secure image Steganography using LSB, DCT and compression techniques on raw images, in: Proceedings of IEEE 3rd International Conference on Intelligent Sensing and Information Processing, ICISIP'05, Bangalore, India, 14-17 Dec. 2005, pp.170-175.

[44] R.T. McKeon, Strange Fourier Steganography in movies, in: Proceedings of the IEEE International Conference on Electro/Information Technology (EIT), 17-20 May 2007, pp. 178-182.

[45] P. Wayner, Disappearing cryptography, 2nd edition, Morgan Kaufmann Publishers, 2002.

[46] C. Manikopoulos, S. Yun-Qing, S. Sui, Z. Zheng, N. Zhicheng, and Z. Dekun, Detection of block DCTbased Steganography in gray-scale images, in: Proceedings of the IEEE Workshop on Multimedia Signal Processing, 9-11 Dec. 2002, pp. 455-458.

[47] N. Provos, Defending against statistical steganalysis, Center for Information Technology Integration, University of Michigan, technical report, February 2001.

[48] N. Provos and P. Honeyman, Detecting Steganography content on the Internet, Center for Information Technology Integration, University of Michigan, technical report, August 31, 2001.

[49] A. Westfeld, F5-A Steganography algorithm: High capacity despite better steganalysis, in: Proceedings of 4th International Workshop on Information Hiding, LNCS 2147, Pittsburgh, USA, April 2001, pp. 289-402.

[61] C.C. Chang and H.W. Tseng, A Steganography method for digital images using side match, Pattern Recognition Letters, 25 (12)(2004)1441-1447.

[62] E. Franz and A. Schneidewind, Adaptive Steganography based on dithering, in: Proceedings of the ACM Workshop on Multimedia and Security, September 20-21, 2004, Magdeburg, Germany, pp.55-62.

[63] R. Böhme and A. Westfeld, Breaking cauchy model-based JPEG Steganography with first order statistics, in: Proceedings of the European Symposium on Research in Computer Security, ESORICS 2004, Valbonne, France, 13th Sept. 2004, LNCS, vol.4194, pp.125-140.

[64] L. Yu, Y. Zhao, R. Ni and Z. Zhu, PM1 Steganography in JPEG images using genetic algorithm, Soft Computing, 14(4)(2009)393-400

[65] C.C. Chang, P. Tsai and M.H. Lin, An adaptive Steganography for index-based images using codeword grouping, Advances in Multimedia Information Processing-PCM, Springer, (4444)(2004)741-748.

[66] H. Hioki, A data embedding method using BPCS principle with new complexity measures, in: Proceedings of Pacific Rim Workshop on Digital Steganography, July 2002, pp. 40-47.

[67] K.B. Raja, S. Sindhu, T.D. Mahalakshmi, S. Akshatha, B.K. Nithin, M. Sarvajith, K.R. Venugopal and L.M. Patnaik, Robust image adaptive Steganography using integer wavelets, in: Proceedings of the 3rd International Conference on Communication Systems Software and Middleware and Workshops, COMSWARE'08, 5-10 Jan. 2008, pp. 514-521.

[68] Y. Srinivasan, High capacity data hiding system using BPCS Steganography, Master Dissertation, Texas Tech. University, USA, December 2004, pp.8, available from: http://etd.lib.ttu.edu/theses/available/etd-05272008-41295018922590/unrestricted/41295018922590.pdf [59] J. Spaulding, H. Noda, M.N. Shirazi and E. Kawaguchi, BPCS Steganography using EZW lossy compressed images, Pattern Recognition Letters, 24(14)(2002)1579-1587.

[70] J. Fridrich, Application of data hiding in digital images, Tutorial for the ISSPA'99, Brisbane, Australia, August 22-25 1999.

[71] Y. Srinivasan, B. Nutter, S. Mitra, B. Phillips and D. Ferris, Secure transmission of medical records using high capacity Steganography, in: Proceedings of the 17th IEEE Symposium on Computer-Based Medical Systems, CBMS'04, 2004, pp.122-127.

[72] E. Kawaguchi and R.O. Eason, Principle and applications of BPCS Steganography, in: Proceedings of SPIE International Symposium on Voice, Video, and Data Communications, 2-4 Nov. 1998, pp. 454-474.

[73] C.C. Lin, W.L. Tai and C.C. Chang, Multilevel reversible data hiding based on histogram modification of difference images, Pattern Recognition, 41(12)(2008)4582-4591.

[74] Y.T. Wu and F.Y. Shih, Genetic algorithm based methodology for breaking the steganalytic systems, IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, (45)(1)(2005)24-41.

[75] S.P. Maity, M.K. Kundu and P.K. Nandi, Genetic algorithm for optimal imperceptibility in image communication through noisy Channel, in: Proceedings of the International Conference on Neural Information Processing (ICONIP '2004), India, 29 October 2004, pp.700-705.

[76] J. Kong, H. Jia, X. Li, Z. Qi, A novel content-based information hiding scheme, in: Proceedings of the International Conference on Computer Engineering and Technology, 22-24 Jan. 2009, vol. 1, pp. 435-440.

[77] M.W. Chao, C.H. Lin, C.W. Yu and T.Y. Lee, A high capacity 3D Steganography algorithm, IEEE Transactions on Visualization and Computer Graphics, 15(2)(2009)274-284.

[78] A. Bogomjakov, C. Gotsman and M. Isenburg, Distortion-free Steganography for polygon meshes, in: Proceedings of Computer Graphics Forum, Eurographics'08, April 2008, vol. 27 (2), pp.637-642.

[79] H. Nakamura and Q. Zhao, Information hiding based on image morphing, in: Proceedings of 22nd International Conference on Advanced Information Networking and Applications Workshops, AINAW, 25- 28 March 2008, pp.1585-1590.

[80] A. M. Zeki and A. A. Manaf, A novel digital watermarking technique based on ISB (Intermediate Significant Bit), World Academy of Science, Engineering and Technology, 48(2009)1080-1087.

[81] Y. H. Yu, C.C. Chang and I.C. Lin, A new Steganography method for color and grayscale image hiding. Computer Vision and Image Understanding, 107(4)(2007)184-194.

[82] M. Drew and S. Bergner, Spatio-chromatic decorrelation for color image compression, Technical Report, School of Computing Science, Simon Fraser University, Vancouver, Canada, 2007, available from : http://fas.sfu.ca/pub/cs/TR/2007/CMPT2007-09.pdf

[83] M. Saenz, R. Oktem, K. Egiazarian and E. Delp, (2000). Color image wavelet compression using vector morphology, in: Proceedings of the European Signal Processing Conference, September 5-8 2000, Tampere, Finland, pp. 5-8.

[84] A. Rodriguez and L. Rowe, Multimedia systems and applications, IEEE Computer, 28 (5)(1995)20- 22.

[85] D. Van Der Weken, M. Nachtegael and E. Kerre, Using similarity measures and homogeneity for the comparison of images, Image and Vision Computing, 22 (9)(2004)595-702.

[86] M. Kutter and F. Petitcolas, A fair benchmark for image watermarking systems, in: Proceedings of Electronic Imaging '99, Security and Watermarking of Multimedia Contents, San Jose, California, U.S.A, 25–27 January 1999, vol. 4557, pp. 225-249.

[87] S.C. Katzenbeisser, Principles of Steganography, in: S. Katzenbeisser and F.A.P Petitcolas, (ed.), Information hiding techniques for Steganography and digital watermarking, Norwood: Artech House, INC, 2000.

[88] J. Fridrich and M. Goljan, Practical steganalysis of digital images-state of the art, in: Proceedings of SPIE Photonics West, Electronic Imaging'02, Security and Watermarking of Multimedia Contents, San Jose, California, January 2002, vol. 4575, pp. 1-14.

[89] A. Martin, G. Sapiro and G. Seroussi, Is image Steganography natural?, IEEE Transactions on Image Processing, 14(12)(2005)2040-2050.

[90] S. Areepongsa, N. Kaewkamnerd, Y.F. Syed and K.R. Rao, Exploring on Steganography for low bit rate wavelet based coder in image retrieval system, in: Proceedings of IEEE TENCON, Kuala Lumpur, Malaysia, 2000, vol.4, pp. 250-255.

[91] P. Kruus, C. Scace, M. Heyman and M. Mundy, A survey of Steganography techniques for image files, Advanced Security Research Journal, V (I) (2004)41-51.

[92] A. Cheddad, J. Condell, K. Curran and P. Mc Kevitt, Skin tone based Steganography in video files exploiting the YCbCr colour space, in: Proceedings of the IEEE International Conference on Multimedia and Expo, Hannover, Germany, June 24-25, 2008, pp.905-909.

[93] A. Cheddad, J. Condell, K. Curran and P. Mc Kevitt, A skin tone detection algorithm for an adaptive approach to Steganography, Signal Processing, 89 (12)(2009) 2455-2478.

[94] A. Nikolaidis and I. Pitas, Region-based image watermarking, IEEE Transactions on Image Processing, 10(11)(2001)1725-1740.

[95] A. Nikolaidis and I. Pitas, Robust watermarking of facial images based on salient geometric pattern matching, IEEE Transactions on Multimedia, 2(4)(2000)172-184.

[96] D.C. Lou and C.H. Sung, A Steganography scheme for secure communications based on the chaos and Euler theorem, IEEE Transactions on Multimedia, 5(4)(2004)501-509.

[97] A. Cheddad, J. Condell, K. Curran and P. Mc Kevitt, Securing information content using new encryption method and Steganography, in: Proceedings of the 3rd IEEE International Conference on Digital Information Management, University of East London, UK, 14-15 Nov. 2008, pp. 554-558.

[98] D.C. Wu and W.H. Tsai, A Steganography method for images by pixel-value differencing, Pattern Recognition Letters, 24 (9-10)(2004)1514-1525.

[99] J. Kodovsky and J. Fridrich, Influence of embedding strategies on security of Steganography methods in the JPEG domain, in: Proceedings of SPIE Electronic Imaging, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X, San Jose, CA, January 28-40, 2008, vol. 5819, pp. 1-14.

[100] Y.S. Chen and R.Z. Wang, steganalysis of reversible contrast mapping watermarking, IEEE Signal Processing Letters, 15(2)(2009)125-128.

[101] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray and S. Vo, A statistical test suite for random and pseudorandom number generators for cryptographic applications, National Institute of Standards and Technology (NIST), special publication 800- 22, Aug. 2008.

[102] L.M. Marvel and C.T. Retter, A methodology for data hiding using images, in: Proceedings of IEEE Military Communications Conference, MILCOM'98, Boston, MA, USA, 18-21 Oct. 1998, pp.1044-1047.

[103] R.J. Anderson and F.A.P. Petitcolas, On the limits of Steganography, IEEE Journal of Selected Areas in Communications, 15(4)(1998)474-481.

[104] P. Bas, Analyse stéganographique d'images numériques : Comparaison de différentes méthodes, Rapport de stage, Laboratoire des Images et des Signaux, University of Joseph Fourier, 23rd June 2004, [in French].

[105] J. Fridrich, M. Goljan and R. Du, Reliable detection of LSB Steganography in grayscale and color images, in: Proceedings of ACM, Special Session on Multimedia Security and Watermarking, Ottawa, Canada, 5th Oct. 2001, pp. 27-40.

[106] I. Avcibas, N. Memon and B. Sankur, Image steganalysis with binary similarity measures, in: Proceedings of the IEEE International Conference on Image Processing, 24-28 June 2002, vol.4, pp.545-548.

[107] P. Civicioglu, M. Alci and E. Besdok, Impulsive noise suppression from images with the noise exclusive filter, EURASIP Journal on Applied Signal Processing, 2004(15)(2004)2444-2440.

[108] R. Bohme and A. Westfeld, Exploiting preserved statistics for steganalysis, Lecture Notes in Computer Science, Springer, 4200/2005(2005)82-95.

[109] M.S. Fu, O.C. Au, Data hiding watermarking for halftone images, IEEE Transactions on Image Processing, 11(4)(2002)477-484.

[110] J.M. Guo, Watermarking in dithered halftone images with embeddable cells selection and inverse halftoning, Signal Processing, 88(5)(2008)1495-1510.

[111] J. Cheng and A.C. Kot, steganalysis of halftone image using inverse halftoning, Signal Processing, 89(5)(2009)1000-1010.

[112] J. Fridrich, M. Goljan and R. Du, Detecting LSB Steganography in color and gray-scale images, IEEE Multimedia, 8(4)(2001)22-28.

[113] T. Pevny and J. Fridrich, Merging Markov and DCT features for multi-class JPEG steganalysis, in: Proceedings of SPIE Electronic Imaging, Photonics West, USA, January 2007, pp. 03-04.

[114] B. Li, Y.Q. Shi and J. Huang, Steganalysis of YASS, in: Proceedings of 10th ACM Workshop on Multimedia and Security, Oxford, United Kingdom, September 22-24 2008, pp. 139-148.

[115] K. Solanki, A. Sarkar and B.S. Manjunath, YASS: Yet another Steganography scheme that resists blind steganalysis, in: Proceedings of the 9th International Workshop on Information Hiding, Saint Malo, France, June 11-14 2007, LNCS, vol. 4557, pp.15-41.