CHAPTER-1

INTRODUCTION

Data hiding is the process of embedding information into a host medium. In general, visual and aural media are preferred due to their wide presence and the tolerance of human perceptual systems involved. Although the general structure of data hiding process does not depend on the host media type, the methods vary depending on the nature of such media.

For instance, image and video data hiding share many common points; however video data hiding necessitates more complex designs as a result of the additional temporal dimension. Therefore, video data hiding continues to constitute an active research area. We propose a temporal synchronization technique to cope with temporal attacks, such as frame drop, insert and repeat.

Data hiding in video sequences is performed in two major ways: bit stream-level and data-level. In bit stream-level, the redundancies within the current compression standards are exploited. Typically, encoders have various options during encoding and this freedom of selection is suitable for manipulation with the aim of data hiding. However, these methods highly rely on the structure of the bit stream; hence, they are quite fragile, in the sense that in many cases they cannot survive any format conversion or transcoding, even without any significant loss of perceptual quality. As a result, this type of data hiding methods is generally proposed for fragile applications, such as authentication. On the other hand, data-level methods are more robust to attacks. Therefore, they are suitable for a broader range of applications. Despite their fragility, the bit stream-based methods are still attractive for data hiding applications

1.1 Problem Definition

During video data hiding the quality of the image cannot be preserved. They cannot survive any format conversion without any significant loss of perceptual quality.

LSB is based on inserting data in the least significant bit of pixels, which lead to slight change on the cover image that is not noticeable to human eye. Since this method can

be easily cracked. It is more vulnerable to attacks. So, the problem definition can be stated as below to overcome the problem defined.

"To hide data efficiently in video and preserve the quality from the external attacks".

1.2 Objective of the project

The main objective of the project is to protect the database from unauthorized and the destructive forces and to increase the security for the data so that the data cannot be easily cracked.

CHAPTER-2

LITERATURE SURVEY

2.1 Introduction

Literature survey is the most important step in software development process..In order to limit perceivable distortion while hiding large amounts of data, hiding schemes must use image adaptive criteria in addition to statistical criteria based on theory. The use of local criteria to choose where to hide data can potentially cause de synchronization of the encoder and decoder. This Synchronization problem is solved by the use of powerful, but simple-to-implement, erasures and errors correcting codes, which also provide robustness against a variety of attacks. Common forward error correction is designed to correct substitution errors only. There is no detection of insertions or deletions. Such systems are usually employed in digital watermarking schemes.

There are few techniques that are able to handle this kind of error. We analyse these techniques, they are namely concatenated coding, dynamic programming, and punctured channel coding .In this project, we propose a new block base selective embedding type data hiding framework that encapsulates forbidden zone data hiding .Forbidden Zone Data Hiding (FZDH) method depends on the Forbidden Zone (FZ) concept, which is defined as the host signal range where no alteration is allowed during data hiding process.

2.2 Existing System

- In special domain, the hiding process such as least significant bit (LSB) replacement is done in special domain, while transform domain methods; hide data in another domain such as wavelet domain.
- Least significant bit (LSB) is the simplest form of Steganography. LSB is based on inserting data in the least significant bit of pixels, which lead to a slight change on the cover image that is not noticeable to human eye. Since this method can be easily cracked, it is more vulnerable to attacks.
- LSB method has intense affects on the statistical information of image like histogram. Attackers could be aware of a hidden communication by just checking the Histogram of an image.

2.3 Proposed System

- Data hiding in video sequences is performed in two major ways: bit stream-level and data-level.
- In this paper, we propose a new block-based selective embedding type data hiding framework that encapsulates Forbidden Zone Data Hiding (FZDH)
- By means of simple rules applied to the frame markers, we introduce certain level of robustness against frame drop, repeat and insert attacks.

2.4 Functional Requirements

Functional Requirements describe the interactions between the system and its environment independent of its implementation. The environment includes the user and any other external system with which the system interacts.

Functional requirements of the system are as follows:

1. User needs to hide his/her secret information in a video file.
2. The secret information may be a file, text or any other file.
3. User should extract the actual information from cover file.
4. There should not be any frame drop in video.

2.5 Advantages of proposed System

- User cannot find the original data.
- It is not easily cracked.
- To increase the Security .
- We can hide more than bit.

CHAPTER-3

ANALYSIS

3.1 Introduction

Analysis is the process of examining the aspects of a project in details. The project analysis model describes a project planning problems and solutions. It is used to evaluate the potential merits of an investment or to objectively assess the value of a business or asset. Problem analysis can be used to analyse an existing situation, understand the problems that are preventing the organisation progressing, and generate a range of possible improvement opportunities. It can also be used to assess and scope an opportunity, and determine the likely benefits of seizing it. The analysis phase is used to ensure that everyone understands the vision of the project; it also defines a clear scope. Project analysis plays an important role in the design, execution, feasibility and evaluation of projects.

3.2 Hardware and Software Specifications

Hardware Requirements:

- Hard Disk : 40 GB.
- System : Pentium IV 2.4 GHz.
- Monitor : 15 VGA Colour.
- Mouse : Logitech.
- Ram : 256 Mb.

Software Requirements:

- Operating system : Windows XP Professional
- Front End : JAVA.
- Tool : Eclipse 3.3

3.3 Software Environment

3.3.1 Java Technology

Java technology is both a programming language and a platform.

3.3.2 The Java Programming Language

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

- Simple
- Architecture neutral
- Object oriented
- Portable
- Distributed
- High performance
- Interpreted
- Multithreaded
- Robust
- Dynamic
- Secure

With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called Java byte codes —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.

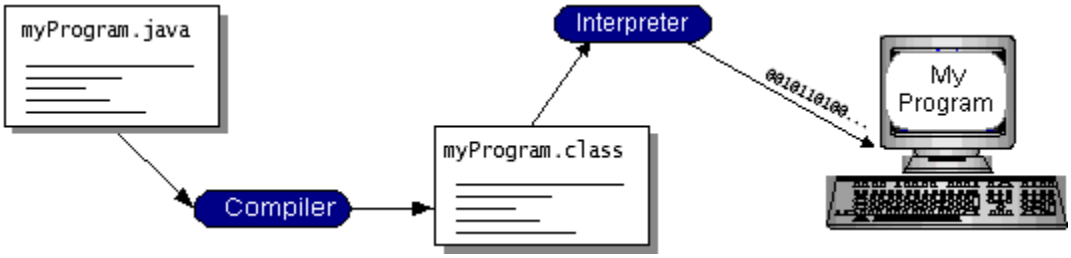


Fig: 3.1 Java Architecture

You can think of Java byte codes as the machine code instructions for the *Java* Virtual Machine (Java VM). Every Java interpreter, whether it's a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make "write once, run anywhere" possible. You can compile your program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.

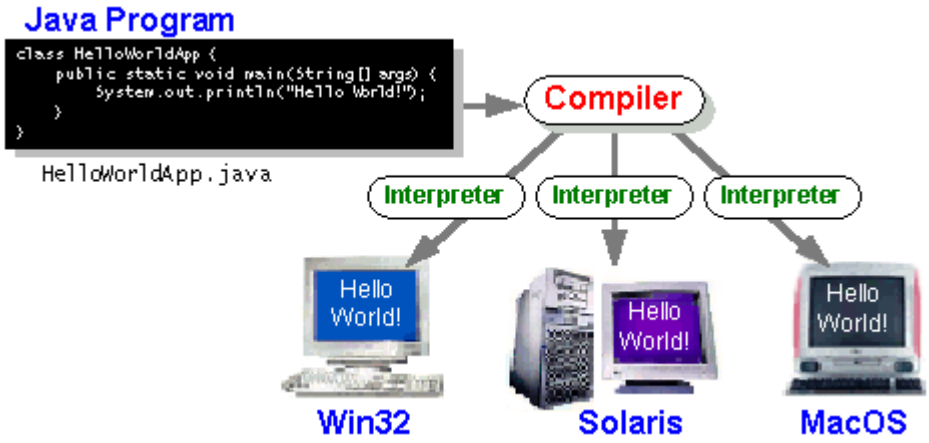


Fig: 3.2 Compilation Structure

3.3.3 The Java Platform

A platform is the hardware or software environment in which a program ruzs1ns. We've already mentioned some of the most popular platforms like Windows 2000, Linux,

Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

The Java platform has two components:

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

You've already been introduced to the Java VM. It's the base for the Java platform and is ported onto various hardware-based platforms.

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces; these libraries are known as packages. The next section, What Can Java Technology Do? Highlights what functionality some of the packages in the Java API provide.

The following figure depicts a program that's running on the Java platform. As the figure shows, the Java API and the virtual machine insulate the program from the hardware.

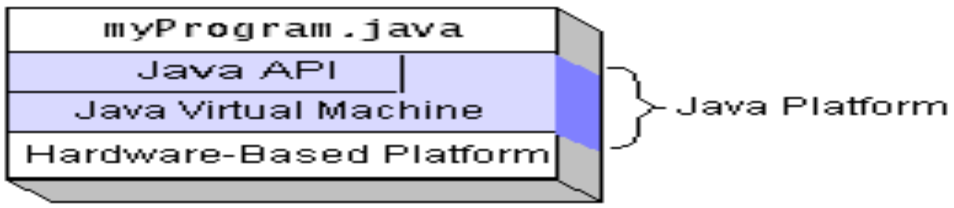


Fig 3.3 Java Platform

Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring performance close to that of native code without threatening portability.

3.3.4 What Can Java Technology Do?

The most common types of programs written in the Java programming language are applets and applications. If you've surfed the Web, you're probably already familiar with applets. An applet is a program that adheres to certain conventions that allow it to run within a Java-enabled browser.

However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java programming language is also a powerful software platform. Using the generous API, you can write many types of programs.

An application is a standalone program that runs directly on the Java platform. A special kind of application known as a server serves and supports clients on a network. Examples of servers are Web servers, proxy servers, mail servers, and print servers. Another specialized program is a servlet. A servlet can almost be thought of as an applet that runs on the server side. Java Servlets are a popular choice for building interactive web applications, replacing the use of CGI scripts. Servlets are similar to applets in that they are runtime extensions of applications. Instead of working in browsers, though, servlets run within Java Web servers, configuring or tailoring the server.

How does the API support all these kinds of programs? It does so with packages of software components that provides a wide range of functionality. Every full implementation of the Java platform gives you the following features:

- **The essentials**: Objects, strings, threads, numbers, input and output, data structures, system properties, date and time, and so on.
- **Applets**: The set of conventions used by applets.
- **Networking**: URLs, TCP (Transmission Control Protocol), UDP (User Data gram Protocol) sockets, and IP (Internet Protocol) addresses.
- **Internationalization**: Help for writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the appropriate language.
- **Security**: Both low level and high level, including electronic signatures, public and private key management, access control, and certificates.
- **Software components**: Known as JavaBeans™, can plug into existing component architectures.
- **Object serialization**: Allows lightweight persistence and communication via Remote Method Invocation (RMI).
- **Java Database Connectivity (JDBC™)**: Provides uniform access to a wide range of relational databases.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more. The following figure depicts what is included in the Java 2 SDK.

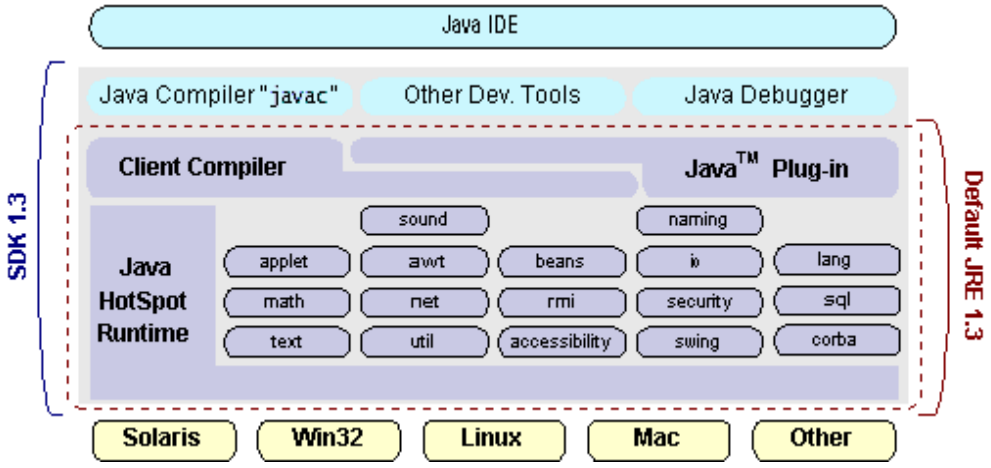


Fig 3.4 Java IDE

How Will Java Technology Change My Life?

We can't promise you fame, fortune, or even a job if you learn the Java programming language. Still, it is likely to make your programs better and requires less effort than other languages. We believe that Java technology will help you do the following:

- **Get started quickly**: Although the Java programming language is a powerful object-oriented language, it's easy to learn, especially for programmers already familiar with C or C++.
- **Write less code**: Comparisons of program metrics (class counts, method counts, and so on) suggest that a program written in the Java programming language can be four times smaller than the same program in C++.
- **Write better code**: The Java programming language encourages good coding practices, and its garbage collection helps you avoid memory leaks. Its object orientation, its JavaBeans component architecture, and its wide-ranging, easily extendible API let you reuse other people's tested code and introduce fewer bugs.

- **Develop programs more quickly**: Your development time may be as much as twice as fast versus writing the same program in C++. Why? You write fewer lines of code and it is a simpler programming language than C++.
- **Avoid platform dependencies with 100% Pure Java**: You can keep your program portable by avoiding the use of libraries written in other languages. The 100% Pure Java™Product Certification Program has a repository of historical process manuals, white papers, brochures, and similar materials online.
- **Write once, run anywhere**: Because 100% Pure Java programs are compiled into machine-independent byte codes, they run consistently on any Java platform.
- **Distribute software more easily**: You can upgrade applets easily from a central server. Applets take advantage of the feature of allowing new classes to be loaded "on the fly," without recompiling the entire program.

3.4 ODBC

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for application developers and database systems providers. Before ODBC became a *de facto* standard for Windows programs to interface with database systems, programmers had to use proprietary languages for each database they wanted to connect to. Now, ODBC has made the choice of the database system almost irrelevant from a coding perspective, which is as it should be. Application developers have much more important things to worry about than the syntax that is needed to port their program from one database to another when business needs suddenly change.

Through the ODBC Administrator in Control Panel, you can specify the particular database that is associated with a data source that an ODBC application program is written to use. Think of an ODBC data source as a door with a name on it. Each door will lead you to a particular database. For example, the data source named Sales Figures might be a SQL Server database, whereas the Accounts Payable data source could refer to an Access database. The physical database referred to by a data source can reside anywhere on the LAN.

The ODBC system files are not installed on your system by Windows 95. Rather, they are installed when you setup a separate database application, such as SQL Server Client or Visual Basic 4.0. When the ODBC icon is installed in Control Panel, it uses a file called

ODBCINST.DLL. It is also possible to administer your ODBC data sources through a stand-alone program called ODBCADM.EXE. There is a 16-bit and a 32-bit version of this program and each maintains a separate list of ODBC data sources. From a programming perspective, the beauty of ODBC is that the application can be written to use the same set of function calls to interface with any data source, regardless of the database vendor. The source code of the application doesn't change whether it talks to Oracle or SQL Server. We only mention these two as an example. There are ODBC drivers available for several dozen popular database systems. Even Excel spreadsheets and plain text files can be turned into data sources. The operating system uses the Registry information written by ODBC Administrator to determine which low-level ODBC drivers are needed to talk to the data source (such as the interface to Oracle or SQL Server). The loading of the ODBC drivers is transparent to the ODBC application program. In a client/server environment, the ODBC API even handles many of the network issues for the application programmer.

The advantages of this scheme are so numerous that you are probably thinking there must be some catch. The only disadvantage of ODBC is that it isn't as efficient as talking directly to the native database interface. ODBC has had many detractors make the charge that it is too slow. Microsoft has always claimed that the critical factor in performance is the quality of the driver software that is used. In our humble opinion, this is true. The availability of good ODBC drivers has improved a great deal recently. And anyway, the criticism about performance is somewhat analogous to those who said that compilers would never match the speed of pure assembly language. Maybe not, but the compiler (or ODBC) gives you the opportunity to write cleaner programs, which means you finish sooner. Meanwhile, computers get faster every year.

3.5 JDBC

In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs. This consistent interface is achieved through the use of In an effort to set an "plug-in" database connectivity modules, or drivers. If a database vendor wishes to have JDBC support, he or she must provide the driver for each platform that the database and Java run on.

To gain a wider acceptance of JDBC, Sun based JDBC's framework on ODBC. As you discovered earlier in this chapter, ODBC has widespread support on a variety of platforms. Basing JDBC on ODBC will allow vendors to bring JDBC drivers to market much faster than developing a completely new connectivity solution.

JDBC was announced in March of 1996. It was released for a 90 day public review that ended June 8, 1996. Because of user input, the final JDBC v1.0 specification was released soon after.

The remainder of this section will cover enough information about JDBC for you to know what it is about and how to use it effectively. This is by no means a complete overview of JDBC. That would fill an entire book.

3.5.1 JDBC Goals

Few software packages are designed without goals in mind. JDBC is one that, because of its many goals, drove the development of the API. These goals, in conjunction with early reviewer feedback, have finalized the JDBC class library into a solid framework for building database applications in Java.

The goals that were set for JDBC are important. They will give you some insight as to why certain classes and functionalities behave the way they do. The eight design goals for JDBC are as follows:

1. **SQL Level API**

   The designers felt that their main goal was to define a SQL interface for Java. Although not the lowest database interface level possible, it is at a low enough level for higher-level tools and APIs to be created. Conversely, it is at a high enough level for application programmers to use it confidently. Attaining this goal allows for future tool vendors to "generate" JDBC code and to hide many of JDBC's complexities from the end user.

2. **SQL Conformance**

   SQL syntax varies as you move from database vendor to database vendor. In an effort to support a wide variety of vendors, JDBC will allow any query statement

to be passed through it to the underlying database driver. This allows the connectivity module to handle non-standard functionality in a manner that is suitable for its users.

3. **JDBC must be implemental on top of common database interfaces**
   The JDBC SQL API must "sit" on top of other common SQL level APIs. This goal allows JDBC to use existing ODBC level drivers by the use of a software interface. This interface would translate JDBC calls to ODBC and vice versa.

4. **Provide a Java interface that is consistent with the rest of the Java system**
   Because of Java's acceptance in the user community thus far, the designers feel that they should not stray from the current design of the core Java system.

5. **Keep it simple**
   This goal probably appears in all software design goal listings. JDBC is no exception. Sun felt that the design of JDBC should be very simple, allowing for only one method of completing a task per mechanism. Allowing duplicate functionality only serves to confuse the users of the API.

6. **Use strong, static typing wherever possible**
   Strong typing allows for more error checking to be done at compile time; also, less error appear at runtime.

7. **Keep the common cases simple**
   Because more often than not, the usual SQL calls used by the programmer are simple SELECT's, INSERT's, DELETE's and UPDATE's, these queries should be simple to perform with JDBC. However, more complex SQL statements should also be possible. Finally we decided to proceed the implementation using Java Networking.

And for dynamically updating the cache table we go for MS Access database.

Java ha 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
- Robust
- Dynamic
- Secure

Java is also unusual in that each Java program is both compiled and interpreted. With a compile you translate a Java program into an intermediate language called Java byte codes the platform-independent code instruction is passed and run on the computer.

Compilation happens just once; interpretation occurs each time the program is executed. The figure illustrates how this works.

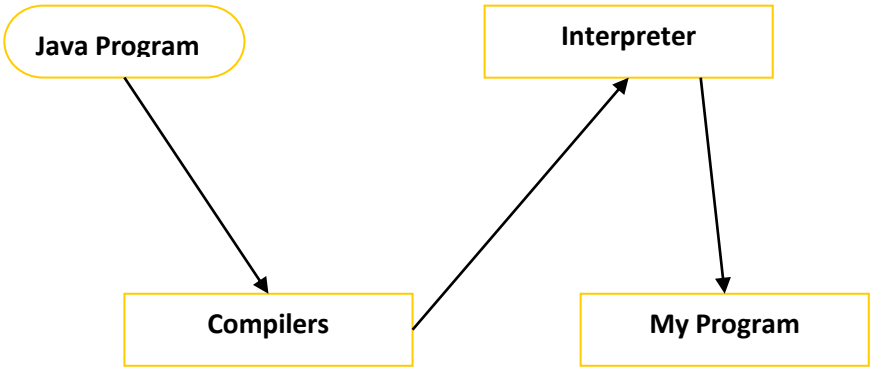


Fig: 3.5 Java Compilation and Interpretation

You can think of Java byte codes as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter, whether it's a Java development tool or a Web browser that can run Java applets, is an implementation of the Java VM. The Java VM can also be implemented in hardware.

Java byte codes help make "write once, run anywhere" possible. You can compile your Java program into byte codes on my platform that has a Java compiler. The byte codes can then be run any implementation of the Java VM. For example, the same Java program can run Windows NT, Solaris, and Macintosh.

CHAPTER-4
DESIGN

4.1 Introduction

Project design is a major first step towards a successful project. A project design is a strategic organization of ideas, materials and processes for the purpose of achieving a goal.

The project design includes everything from who is responsible for completing the project to a description of the project, its goals, outcomes and objectives. It describes when these goals, outcomes and objectives will be reached, and the major deliverables, products or features that will be completed. The project design also estimates the budget and outlines how to monitor and evaluate progress. It's a systematic series of steps that helps you to define, plan and produce a product you're building — in our case, an app. It allows you to be efficient, transparent and focused on creating the best product possible.

4.2 UML Diagrams

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.

- UML stands for **Unified Modeling Language**.
- UML is different from the other common programming languages such as C++, Java, COBOL, etc.
- UML is a pictorial language used to make software blueprints.
- UML can be described as a general purpose visual modeling language to visualize, specify, construct, and document software system.
- Although UML is generally used to model software systems, it is not limited within this boundary. It is also used to model non-software systems as well. For example, the process flows in a manufacturing unit, etc.

UML is not a programming language but tools can be used to generate code in various languages using UML diagrams.

4.3 Why use UML in projects

The main aim of UML is to define a standard way to visualize the way a system has been designed. It is quite similar to blueprints used in other fields of engineering. UML is not a programming language; it is rather a visual language. We use UML diagrams to portray the behaviour and structure of a system.

UML diagrams can be used as a way to visualize a project before it takes place or as documentation for a project afterward. But the overall goal of UML diagrams is to allow teams to visualize how a project is or will be working, and they can be used in any field, not just software engineering.UML diagrams are really helpful in designing and maintaining an application. An UML diagram can show the structure and logic of an object oriented program.

UML is a standardized general purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating for creating models of object oriented computer software. In its current form UML is comprised of two major components. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

The Unified Modeling Language is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

4.4 DIAGRAMS

4.4.1 Use Case Diagram

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analyzed to gather its functionalities, use cases are prepared and actors are identified. The purpose of use case diagram is to capture the dynamic aspect of a system.

A Use Case diagram is a graphic depiction of the interactions among the elements of a system. A Use Case is a methodology used in system analysis to identify, clarify, and organize system requirements .It is a way to capture the system's functionality and requirements in UML diagrams .It captures the dynamic behaviour of a live system .A Use Case diagram consists of a use case and an actor. A Use Case represents a functionality of a system, a component, a package, or a class.

`

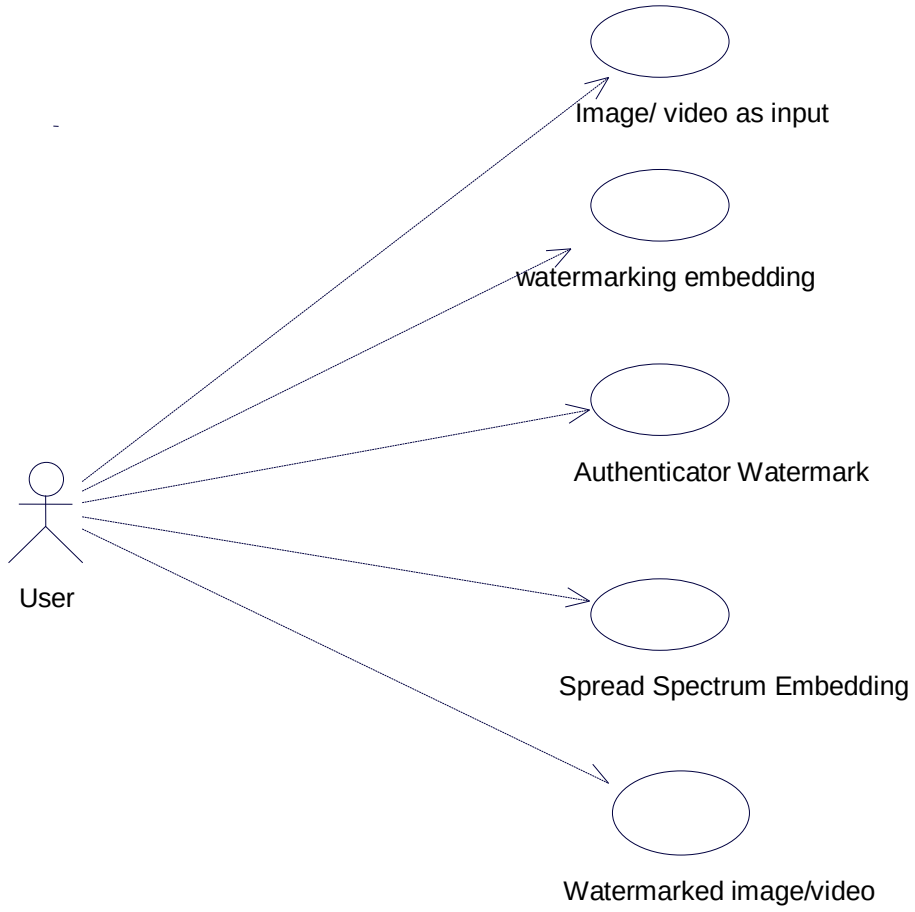


Fig 4.1: Usecase Diagram

4.4.2 Dataflow Diagram

A data-flow diagram (DFD) is a way of representing a flow of a data of a process or a system (usually an information system). The DFD also provides information about modelling tools.

It includes data inputs and outputs, data stores, and the various sub processes the data moves DFD are built using standardized symbols and notation to describe various entities and their relationships.

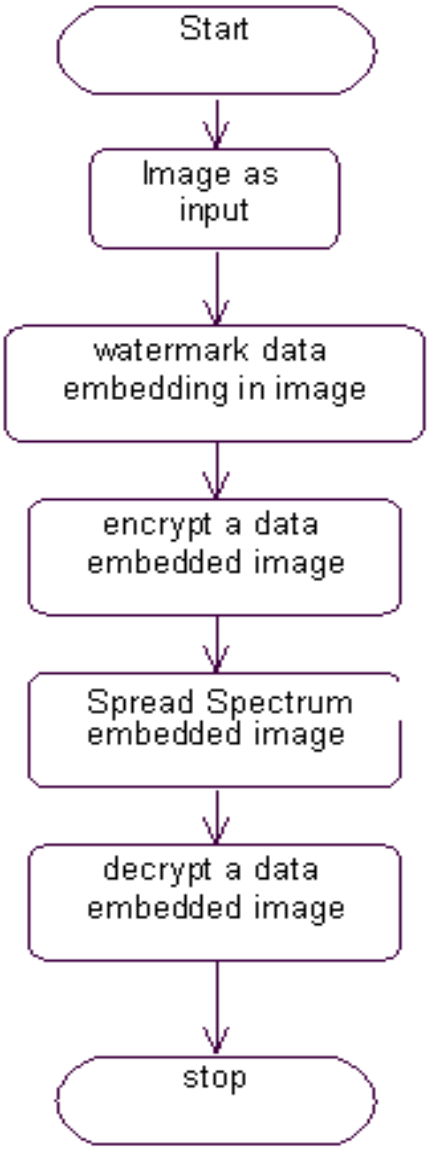


Fig:4.2 Dataflow Diagram

4.4.3 Object Diagram:

An Object Diagram is a UML structural diagram that shows the instances of the classifiers in models. An object diagram is a graph of instances, including objects and data values. A static object diagram is an instance of a class diagram .It shows a snapshot of the detailed

state of a system at a point in time, while class diagrams show classes; object diagrams display instances of classes.

It is used to describe the static aspect of a system. It is used to represent an instance of a class .It can be used to perform forward and reverse engineering on systems. It is used to understand the behaviour of an object. An object model diagram shows the interaction between objects at some point, during run time. An object diagram focuses on the attributes of a set of objects and how those objects relate to each other. An object is an instance of a class, and may be called a class instance or class object. An object is a generic thing while an instance is a single object that has been created in memory.

Usually an instance will have values assigned to its properties that differentiate it from other instances of the type of object.

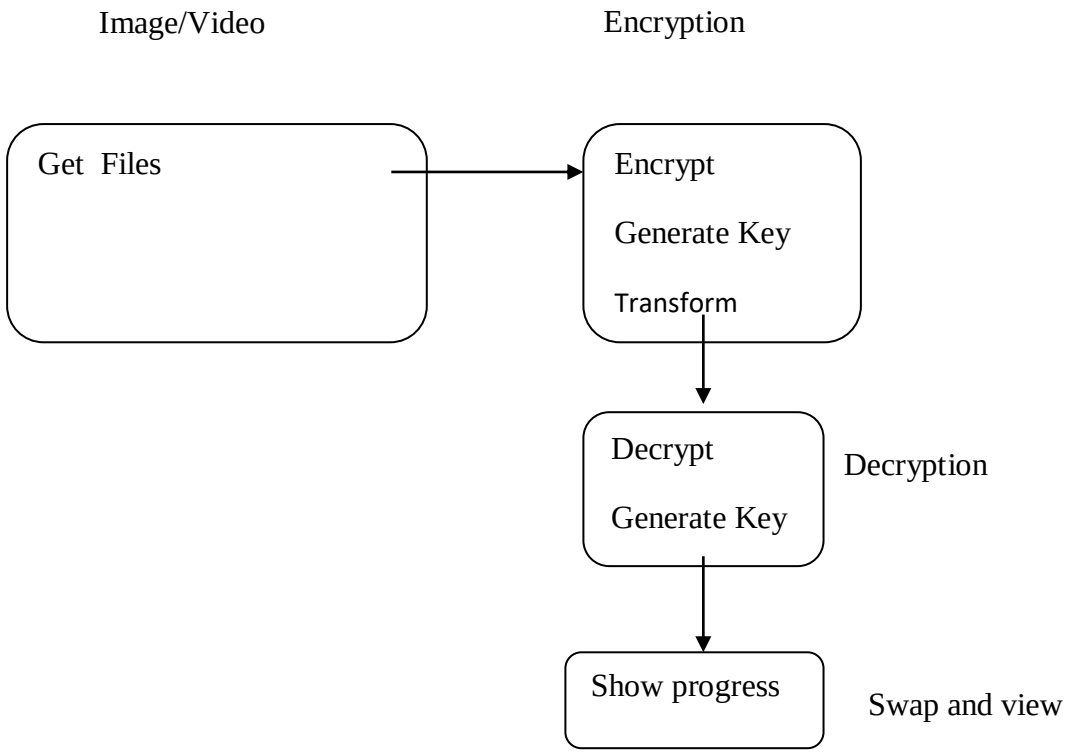


Fig: 4.3 Object Diagram

4.4.4 Sequence Diagram

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place.

A sequence diagram is an interaction diagram that shows how processes operate with zone another and in what order. It is a construct of message sequence chart. A sequence diagram shows object interactions arranged in time sequence. It has two dimensions, Vertical dimension represent time and Horizontal dimension represent different objects. The vertical line called Object Life Line that represents the objects existing during the interaction. The order in which messages appear is shown top to bottom. Each message is labelled with message name.

A sequence diagram is an interaction diagram that shows how processes operate with one another and in what order. It is a construct of message sequence chart. A sequence diagram shows object interactions arranged in time sequence. It has two dimensions, Vertical dimension represent time and horizontal dimension represent different objects. The vertical line called Object Life Line that represents the objects existing during the interaction. The order in which messages appear is shown top to bottom.

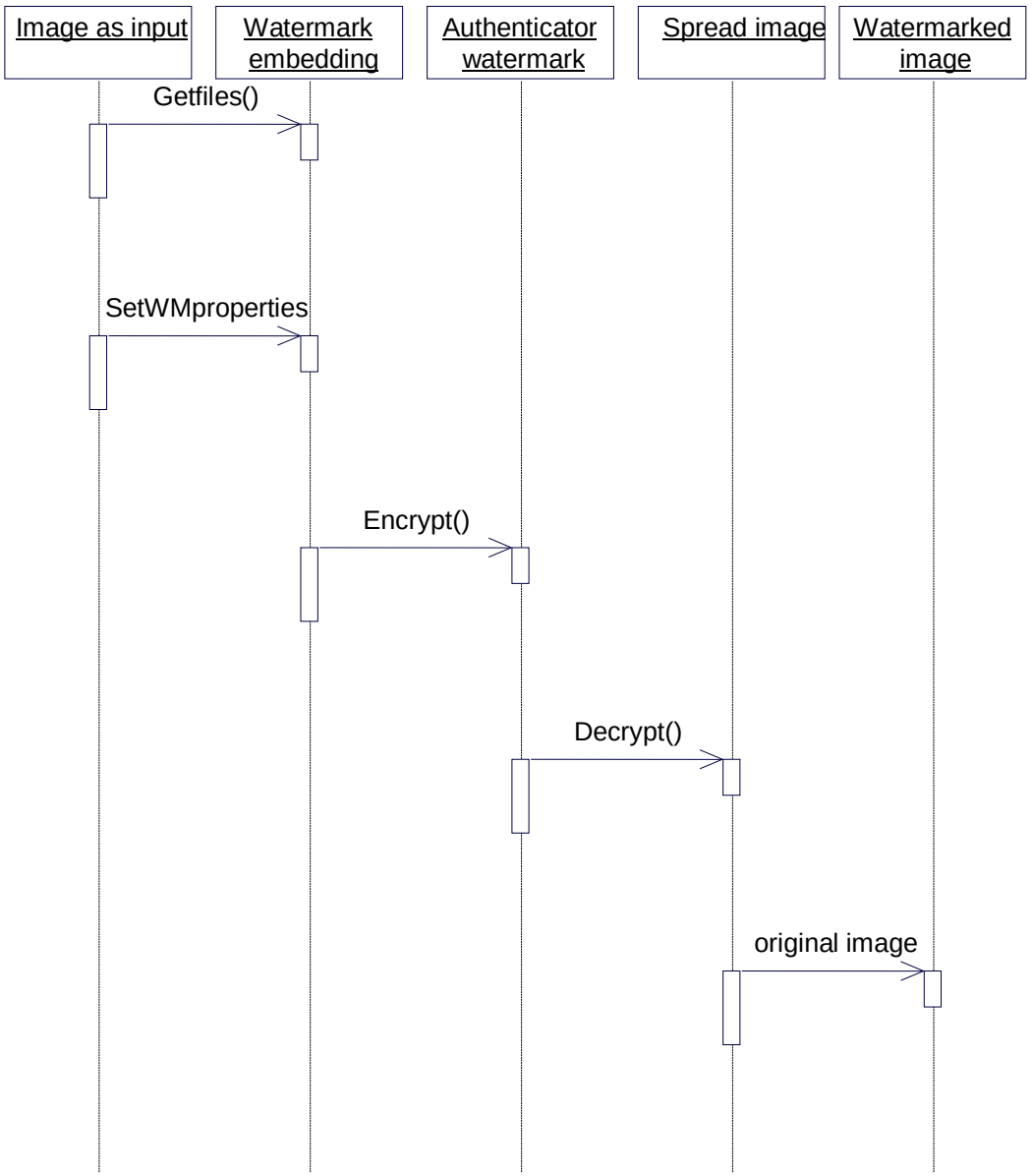


Fig:4.4 Sequence Diagram

4.4.5 Activity Diagram

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

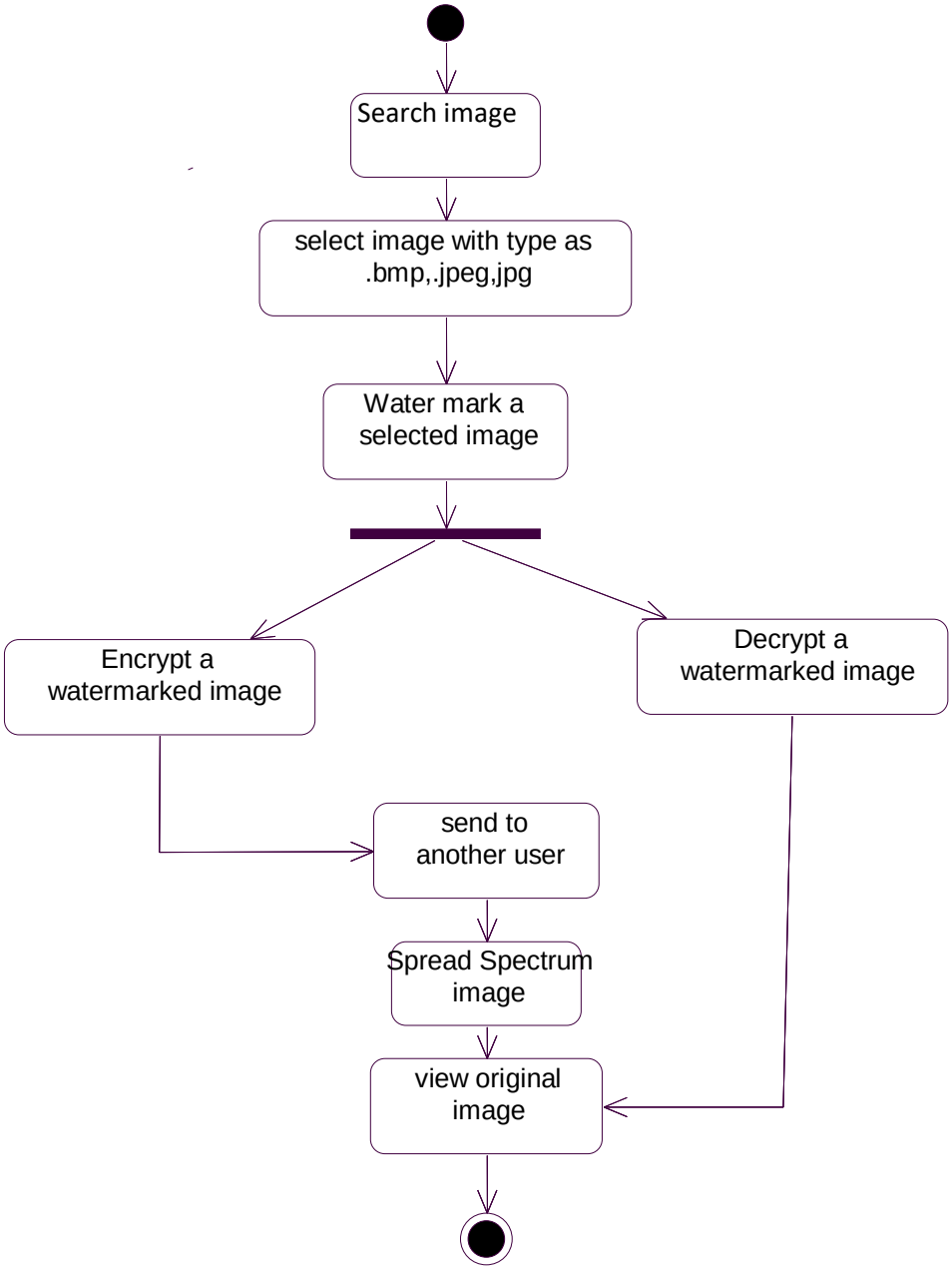


Fig: 4.5 Activity Diagram

4.5 Input Design

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

Objectives

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the userwill not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

4.6 Output Design

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3 .Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

4.7 System Study

4.7.1 Feasibility Study

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

4.7.1.1 Economic Feasibility

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pourinto the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

4.7.1.2 Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

4.7.1.3 Social Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

CHAPTER-5

IMPLEMENTATION

5.1 Introduction

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

Project implementation is the phase where visions and plans become reality. This is the logical conclusion, after evaluating, deciding, visioning, planning, applying for funds and finding the financial resources of a project. Technical implementation is one part of executing a project.

An implementation plan for a project refers to a detailed description of actions that demonstrate how to implement an activity within the project in the context of achieving project objectives, addressing requirements, and meeting expectations. The implementation phase represents the work done to meet the requirements of the scope of work and fulfil the charter.

5.2 Modules

- Input Module
- Encryption Module
- Decryption Module
- DES

5.3 Modules Description

5.3.1 Input Module:

The Input Module is designed as such a way that the proposed system must be capable of handling any type of data formats, such as if the user wishes to hide any image format then it must be compatible with all usual image formats such as jpg, gif, bmp, it must be also compatible with video formats such as .avi, .flv, .wmf etc.. And also it must be compatible with various document formats, so that the user can be able to user any formats to hide the secret data. The input module that connects an input device, such as a panel or sensor, to a programmable logic controller, a device used to control automated manufacturing processes.

5.3.2 Encryption Module:

In Encryption module, it consists of Key file part, where key file can be specified with the password as a special security in it. Then the user can type the data or else can upload the data also though the browse button, when it is clicked the open file dialog box is opened and where the user can select the secret message. Then the user can select the image or video file through another open file dialog box which is opened when the cover file button is clicked. Where the user can select the cover file and then the Hide button is clicked so that the secret data or message is hidden in cover file using Forbidden Zone Data Hiding Technique.

5.3.3 Decryption Module:

This module is the opposite as such as Encryption module where the Key file should be also specified same as that of encryption part. Then the user should select the encrypted cover file and then should select the extract button so that the hidden message is displayed in the text area specified in the application or else it is extracted to the place where the user specifies it.

5.4 DES:

This module consists of same as Encryption and Decryption part using DES algorithm. The Data Encryption Standard (DES) is a block cipher that uses shared secret

encryption. The Data Encryption Standard is a symmetric-key algorithm for the encryption of digital data. Although its short key length of 56 bits makes it too insecure for modern applications, it has been highly influential in the advancement of cryptography.DES is an implementation of a Feistel Cipher. It uses 16 round Feistel structure. Triple-Des is still in use today but is widely considered a legacy encryption algorithm.DES is inherent insecure, while Triple-DES has much better security characteristics but is still considered problematic. The Data Encryption Standard is an outdated symmetric-key method of data encryption.DES works by using the same key to encrypt and decrypt a message, so both the sender and the receiver must know and use the same private key.

5.5 Module I/O:

Module Input:

We give original content as input with watermark data embedding. We view flipping an edge pixel in binary images as shifting the edge location one pixel horizontally and vertically.

Module Output:

The output of the project is we reconstruct the pixel horizontally and vertically .we can see the original watermarked data and embedding content.

CHAPTER-6

SYSTEM TESTING

6.1 Introduction

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

System Testing is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to the end-to-end system specifications. Usually,the software is only one element of a larger computer-based system.

6.2 TYPES OF TESTS

6.2.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

In computer programming, unit testing is a software testing method by which individuals units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. A unit testing is a level of software testing where

individual units components of software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

6.2.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Integration Testing is a level of software testing where individuals units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in integration testing.

6.2.3 Functional testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is a type of software testing where by the system is tested against the functional requirements specifications. Functions are tested by feeding them input and examining the output. Functional testing ensures that the requirements are properly satisfied by the application. It is important because it always verifies that your system is fixed for release. The functional tests define your working system in a useful manner .A Functional tester holds a lot of importance as able to analyze individual pieces of an application within the context of the entire application.

Functional testing is the process of determining how closely a program matches specifications.It is based on the development of test case scenarios derived from program specifications.

Functional testing is centred on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

6.2.4 System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

System testing is a type of software testing that is performed on a complete integrated system to evaluate the compliance of the system. In system testing, integration testing passed components are taken as input. It is a process of testing the entire system that is fully functional, in order to ensure the system is bound to all the requirements provided by the client in the form of the functional specification or system specification documentation.

6.2.5 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

White Box Testing is testing of a software solutions internal structure,design,and coding.In this type of testing,the code is visible to the tester.It focuses primarily on verifying the flow of inputs and outputs through the application ,improving design and usability, strengthening security.

White Box Testing is a software testing method in which the internal structure/design/ implementation of the item being tested is known to the tester.

6.2.6 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

It is defined as a testing technique in which functionality of the application under test is tested without looking at the internal code structure,implememtation details and knowledge of internal paths of the software.Black Box Testing is a software testing method in which the internal structure/design/implementation of the item being tested is not known to the tester.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

6.2.7 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements. It is a testing technique performed to determine whether or not the software system has meet the requirement specifications.

Acceptance testing is a level of software testing where a system is tested for acceptability. The purpose of this test is to evaluate the systems compliance with the business requirements and assess whether it is acceptable for delivery.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

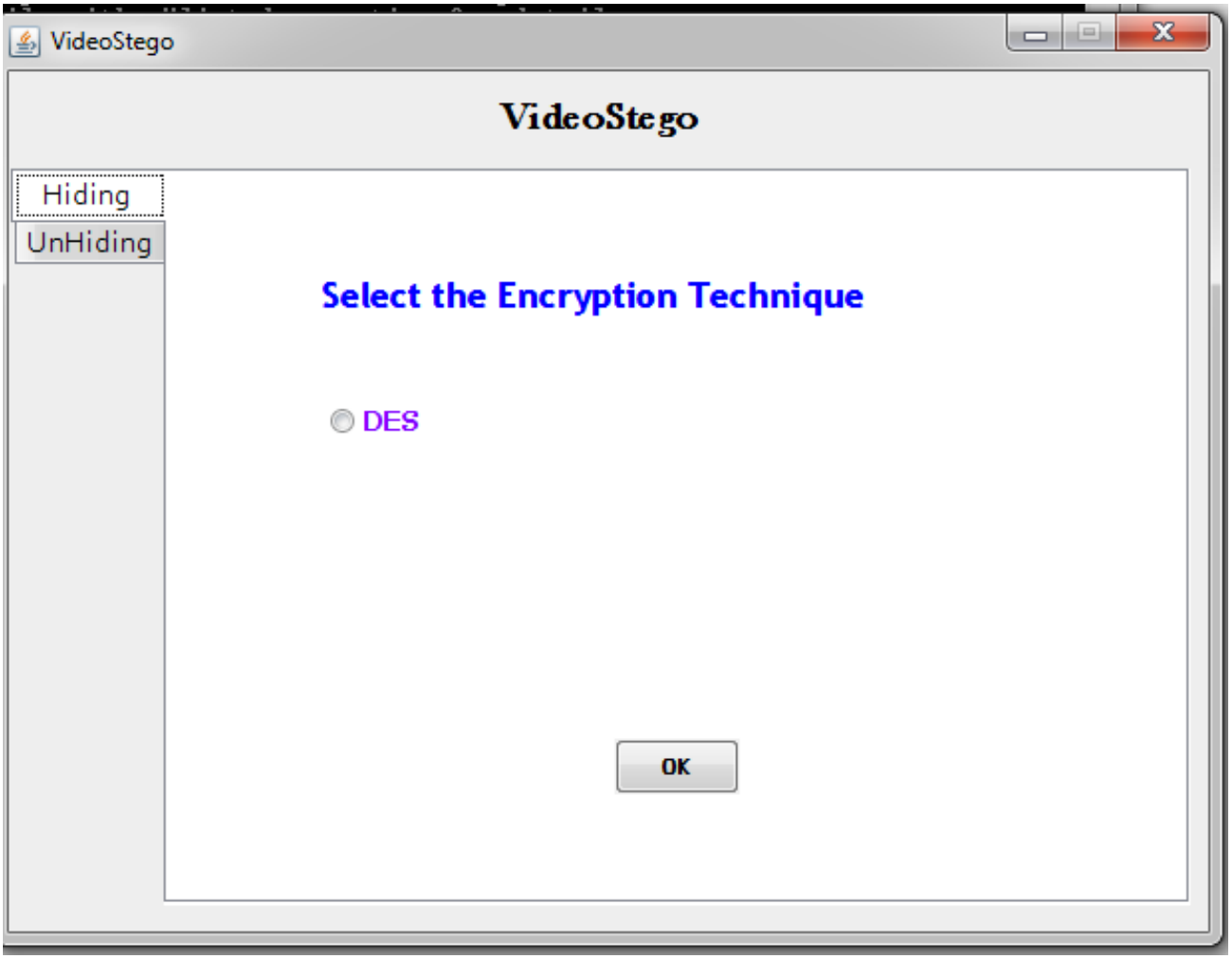
CHAPTER-7

SAMPLE SCREENSHOTS

7.1 Hiding Screenshots

In robust video data hiding using selective embedding and forbidden zone data hiding, we are hiding the secret data in video.

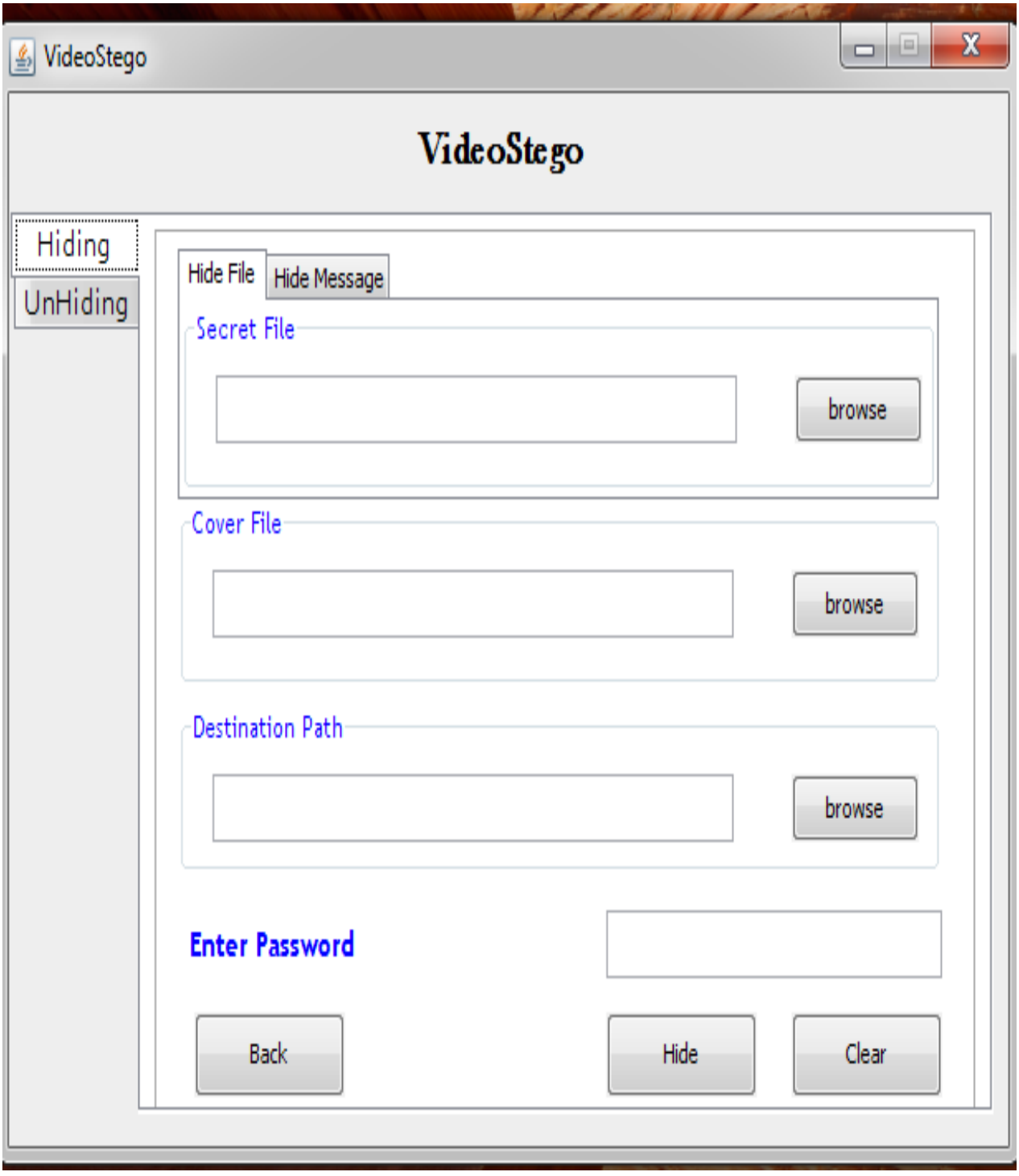
Screen 1: VideoStego Panel



Screen 2: Algorithm Selection

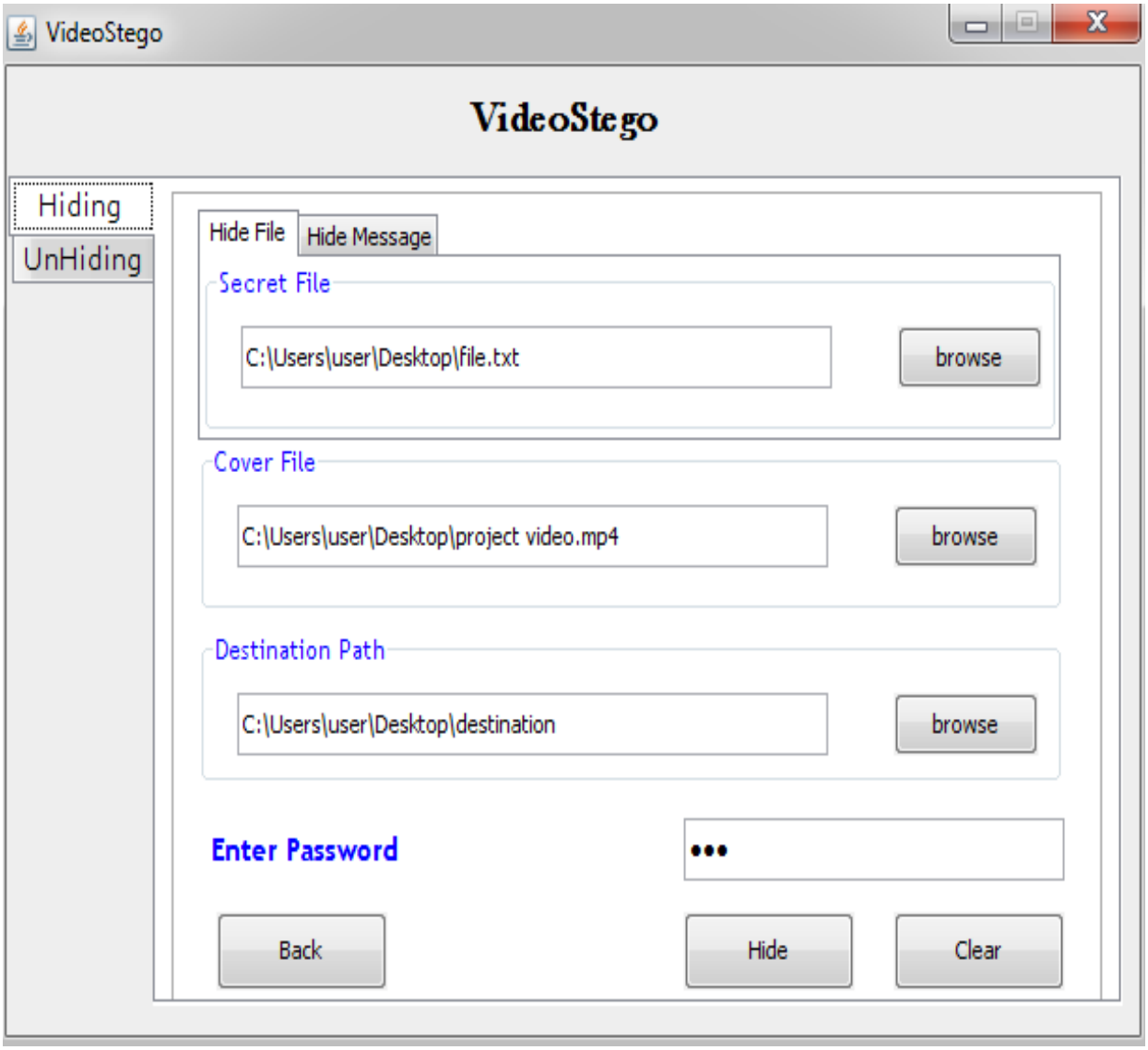
First we select hiding in this process and we have to select the encryption technique i.e.DES(Data Encryption Standard). Data Encryption Standard (DES) is a block cipher that uses shared secret encryption. The Data Encryption Standard is a symmetric-key algorithm for the encryption of digital data. Although it's short key length of 56 bits makes it too insecure for modern applications, it has been highly influential in the advancement of cryptography.DES is an implementation of a Feistel Cipher. It uses 16 round Feistel structure.

Screen 3: Hiding Secret data



In data hiding, while we are hiding the data in a video we need a video, the information which we need to hide in video.

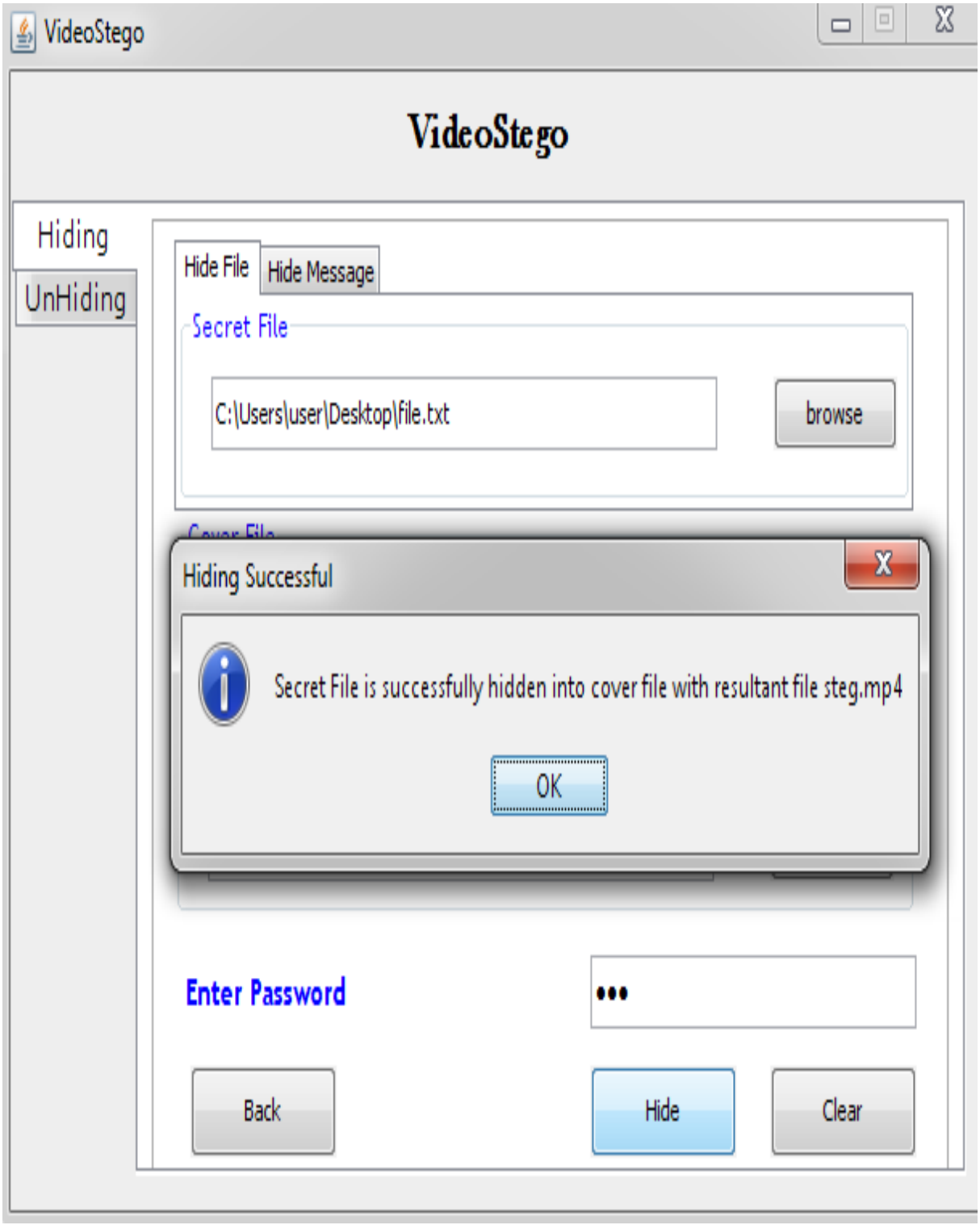
Screen 4: Uploading files



The information which we need to hide the data in a video we should keep it in secret file, and the video will be placed in cover file and we have create one new folder named as destination path.

So after hiding the data in video, that data which is stored in video will be in the destination path. So we have to give password and after clicking hide it automatically hide the data in video.

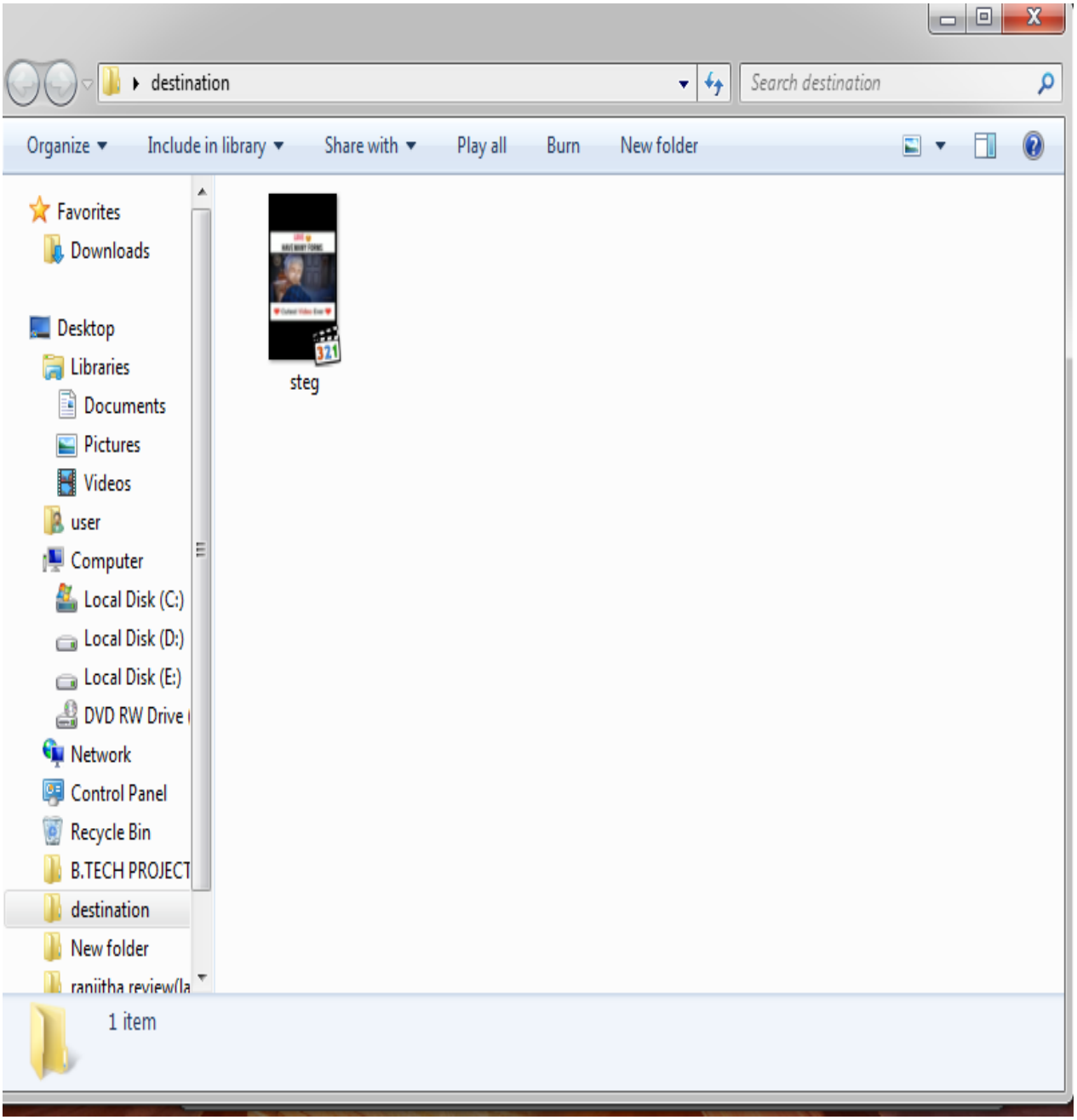
Screen 5: Successful Hiding



-

After hiding it shows that Secret File is successfully hidden into cover page i.e,the secret information is stored video.

Screen 6: Video With hidden data

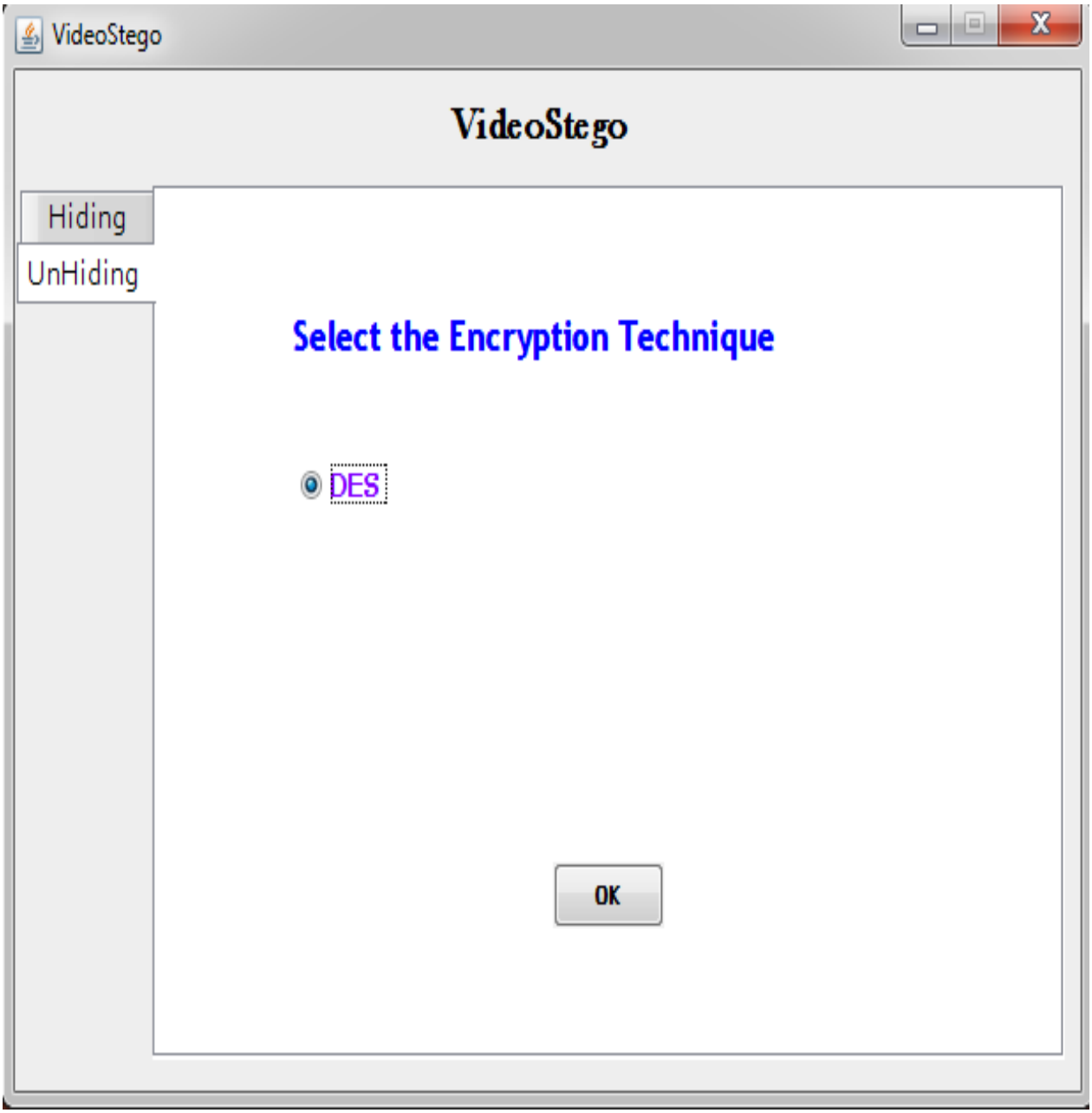


So here it shows the video in destination path the data is successfully hidden...

7.2 Unhiding Screenshots

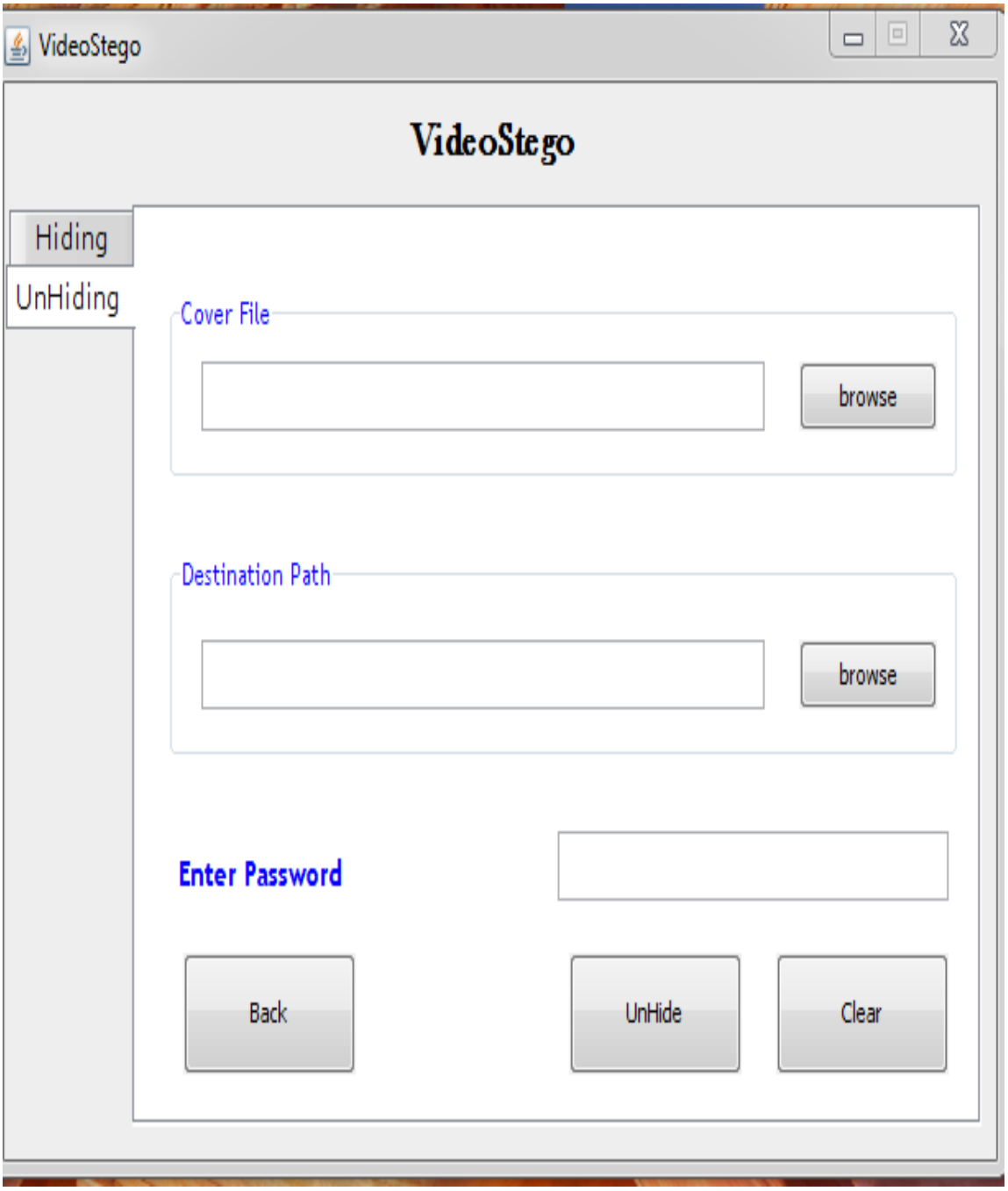
In unhiding the data which we stored in video can be retrieved, by the user after we transfer. The user can retrieve the data by password.

Screen 7: Algorithm Selection



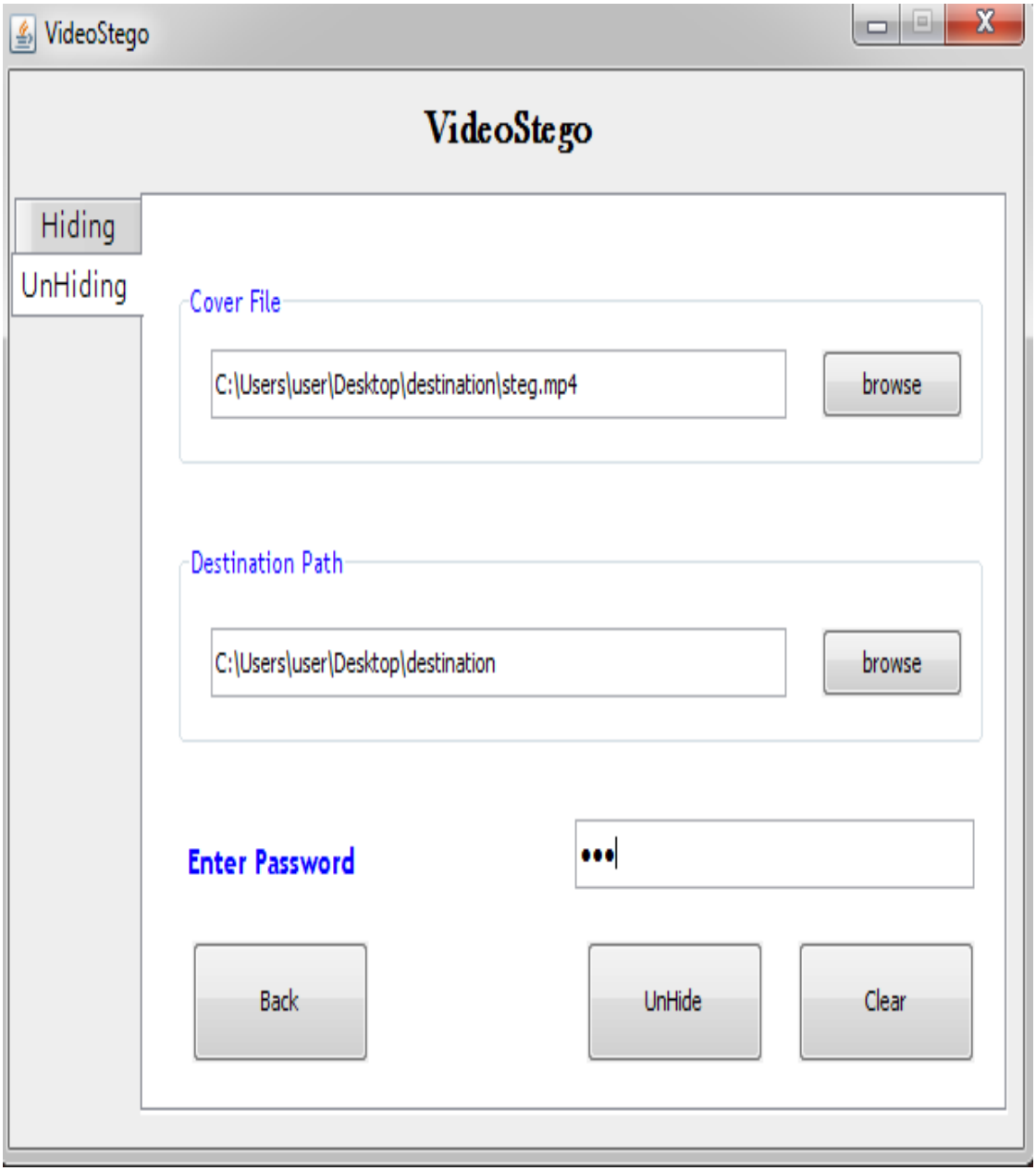
In unhiding the data which we stored in video can be retrieved by the user ,so while we are sending or transferring the secret data to the other person he can retrieve the data by giving our password.

Screen 8: Unhiding Secret data



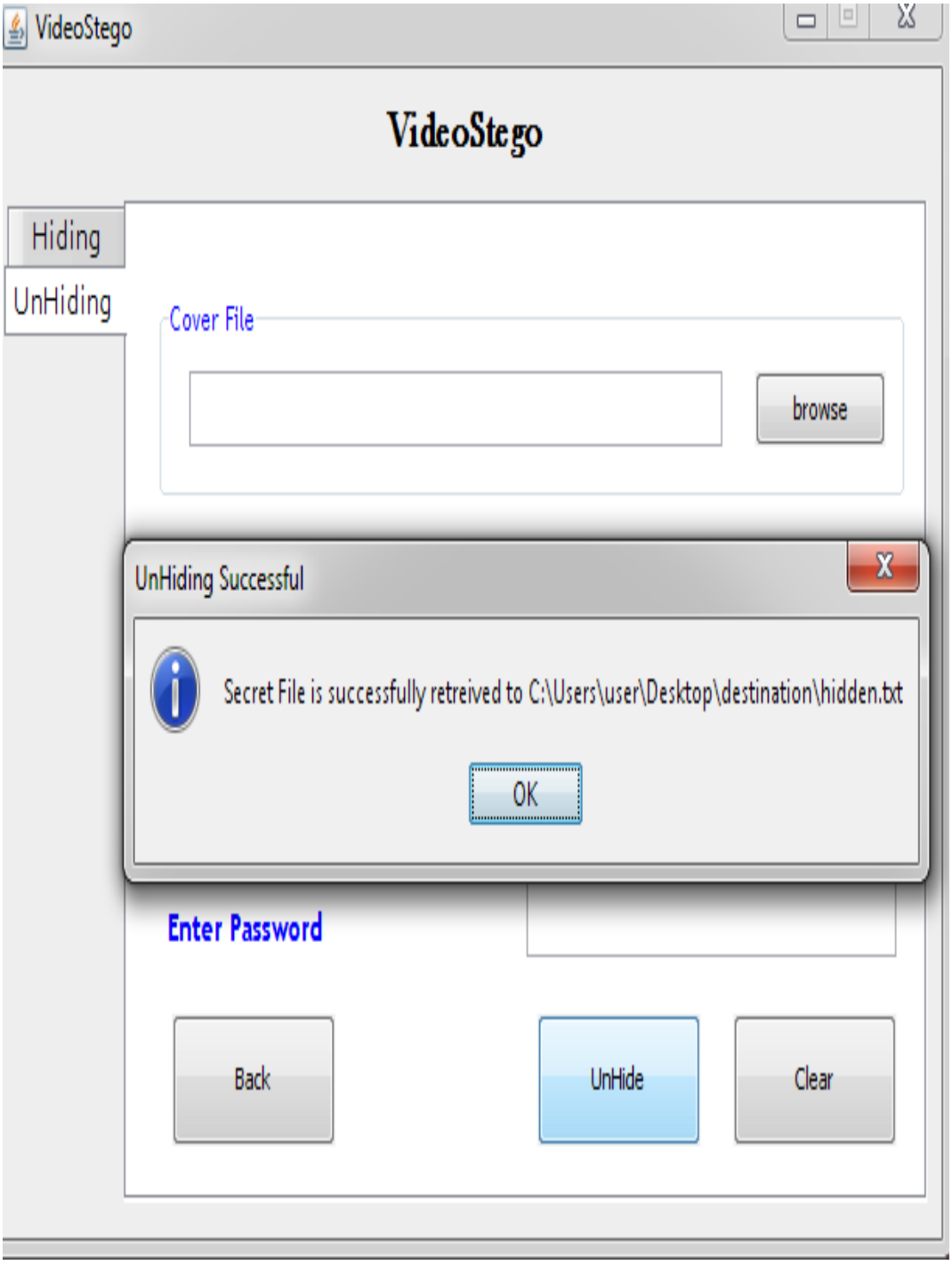
So here the cover file and destination path is used while we are unhiding the data, after transferring the video the user will give the password.

Screen 9: Uploading Videofile



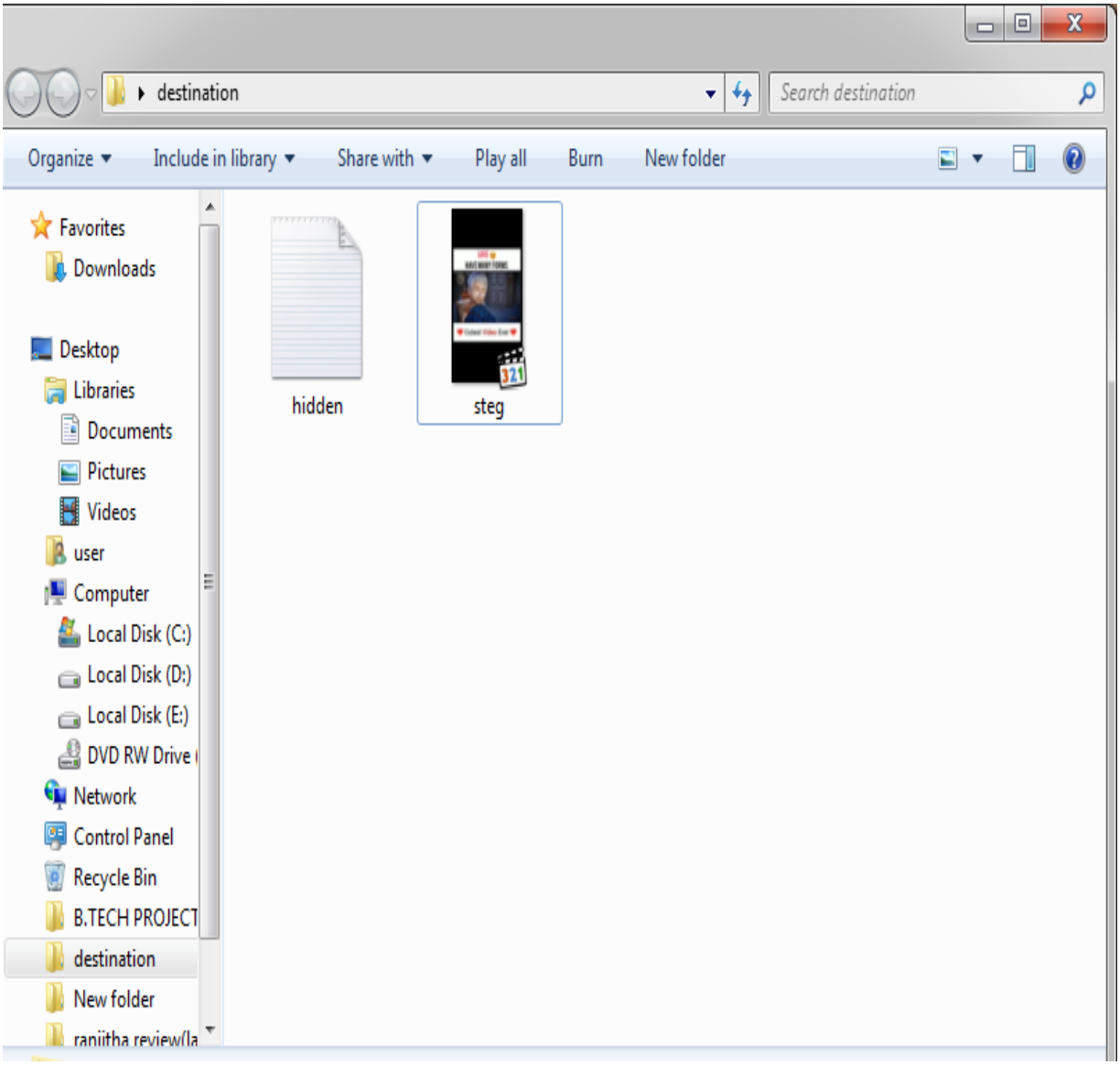
In cover file we have to give the destination path which we have in hiding process and in unhiding the destination path it will take automatically and while they are transferring the secret data in video they used to give the password.

Screen 10: Retreiving Secret data



After giving the password we can retrieve the data which is stored in the video .We get the data in destination folder.

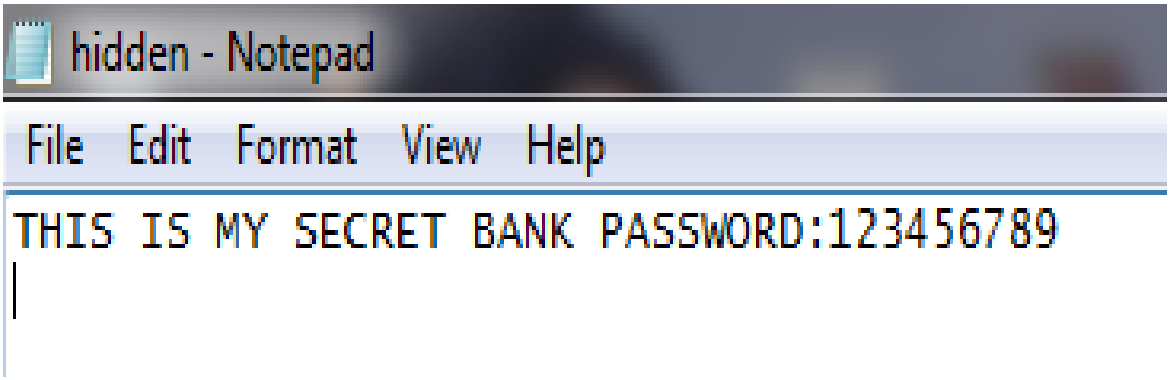
Screen 11: Retreived secret data from video



So here we can see the video which is done in hiding process and also we can see the data which we retrieved in unhiding process. In hiding process we used to give the password to hide the secret data in video, after hiding the data in video we used to transfer the video to whom we are sharing that secret data by giving password.

After sharing the video we used to give the password to whom we shared the data which we stored in video, by knowing the password they can retrieve the secret data.

Screen 12: Secret data Retreived



So this is the password which we stored in video and that was retrieved by the user in unhiding process.

CONCLUSION

In this paper, we propose a new video data hiding framework that makes use of erasure correction capability of frame dropping. The method is also robust to frame manipulation attacks via frame synchronization markers. The framework is tested with MPEG, FLV and scaling and frame-rate conversion attacks.

The proposed framework is tested by typical broadcast material against MPEG frame-rate conversion attacks, as well as other well-known video data hiding methods. The simulation results indicate that the framework can be successfully utilized in video data hiding applications.

BIBLIOGRAPHY

[1] S. K. Kapotas, E. E. Varsaki, and A. N. Skodras, "Data Hiding in H- 264 Encoded Video Sequences," in IEEE 9th Workshop on Multimedia Signal Processing.

[2] A. Sarkar, U. Madhow, S. Chandrasekaran, and B. S. Manjunath,"Adaptive MPEG-2 Video Data Hiding Scheme," in Proceedings of SPIE Security, Steganography, and Watermarking of Multimedia Contents .

[3] K. Solanki, N. Jacobsen, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Robust image-adaptive data hiding using erasure and error correction," IEEE Transactions on Image Processing

[4] M. Schlauweg, D. Profrock, and E. Muller, "Correction of Insertions and Deletions in Selective Watermarking," in IEEE International Conference on Signal Image Technology and Internet Based Systems.

[5] M. Wu, H. Yu, and B. Liu, "Data hiding in image and video I:Fundamental issues and solutions," IEEE Transactions on Image Processing.

[6] M. Wu, H. Yu, and B. Liu, "Data hiding in image and video II:Designs and applications," IEEE Transactions on Image Processing.

[7] E. Esen and A. A. Alatan, "Forbidden zone data hiding," in IEEE International Conference on Image Processing.

[8] B. Chen and G. W. Wornell, "Quantization index modulation: a class of provably good methods for digital watermarking and information embedding," IEEE Transactions on Information Theory.

[9] E. Esen, Z. Doğan, T. K. Ates, and A. A. Alatan, "Comparison of Quantization Index Modulation and Forbidden Zone Data Hiding for Compressed Domain Video Data Hiding".

[10] K. Wong, K. Tanaka, K. Takagi, and Y. Nakajima, "Complete video quality-preserving data hiding," IEEE Transactions on Circuits and Systems for Video Technology.

Sites Referred:

http://java.sun.com

http://www.sourcefordgde.com

http://www.networkcomputing.com/

http://www.roseindia.com/

http://www.java2s.com/

http://www. javadb.com/