NORTHERN ILLINOIS UNIVERSITY

A Thesis Submitted to the

University Honors Program

In Partial Fulfillment of the

Requirements of the Baccalaureate Degree

With Upper Division Honors

Department Of

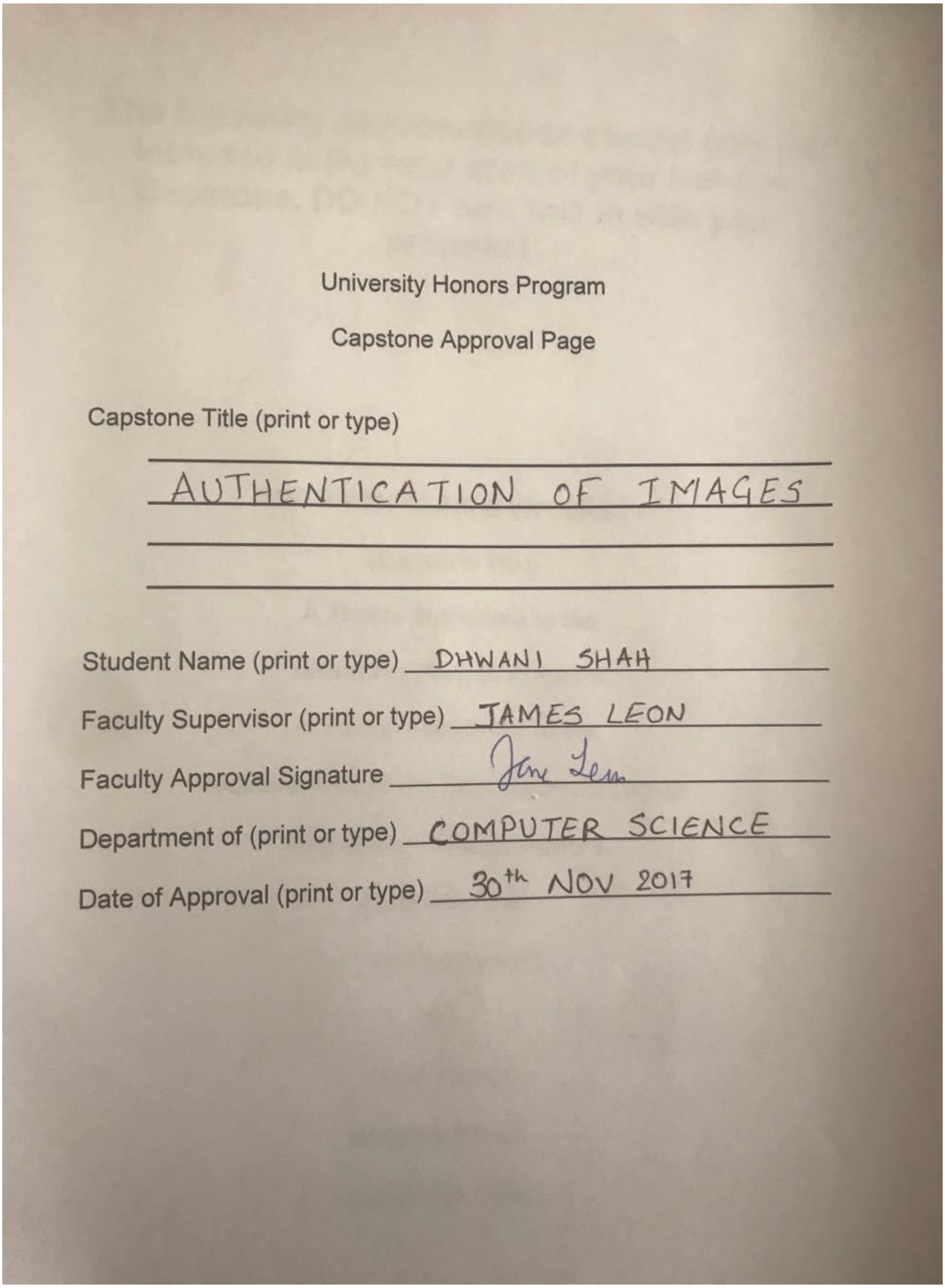
Computer Science

By

Dhwani Shah

DeKalb, Illinois

December 2017



University Honors Program

Capstone Approval Page

Capstone Title (print or type)

AUTHENTICATION OF IMAGES

Student Name (print or type) DHWANI SHAH

Faculty Supervisor (print or type) JAMES LEON

Faculty Approval Signature

Department of (print or type) COMPUTER SCIENCE

Date of Approval (print or type) 30th NOV 2017

HONORS THESIS ABSTRACT

THESIS SUBMISSION FORM

AUTHOR: Dhwani Shah

THESIS TITLE: Authentication of Images

ADVISOR: James Leon

ADVISOR'S DEPARTMENT: Computer Science

DISCIPLINE: Computer Science YEAR: 2017

PAGE LENGTH: 50

ABSTRACT

Now a day’s mobile phones have replaced cameras for capturing images because of its ubiquitous nature. Mobile phones are used for capturing and distributing (WhatsApp, Facebook) images for professional as well as personal uses. So, at some point we need to prove the ownership of the image .It is a kind of copyright that the image was captured by a particular mobile phone camera. Copyright is a form of intellectual property applicable to any expressible form of an idea or information that is substantive and discrete. Copyright prevents others from taking others work for free. It also prevents people from altering the work without permission. The problem arises when you have to prove that you are the owner of the image. Proposing a technique which embeds personal mobile phone numbers along with IMEI number inside the image using invisible watermarking technique. A invisible watermark is a pattern of bits inserted into a digital image file that identifies the file's copyright information.

LIST OF FIGURES

LIST OF SYMBOLS, ABBREVIATIONS AND NOMENCLATURE

DCT – Discrete Cosine Transform

DWT - Discrete Wavelet Transformation

LSB - Least Significant Bit

MSISDN – Mobile Station International Subscriber Directory Number

IMEI - International Mobile Equipment Identity

TABLE OF CONTENTS

CHAPTER1: INTRODUCTION

Before some years back, mobile devices are used for only sending message, playing video/audio, internet browsing etc. Before so but now a day, smart phones are introducing with incredible changes such as image publishing etc. So, there is need to provide protection to images to avoid loss of information like as ownership information. Prevent images from those people which do not have any access permission. By using digital watermark we can embed the information into image. This technique is mostly use to identify the ownership of the copyright of such image. Watermarking means hide digital information in a image data. Digital watermark is use to verify the authentication of image or to show the identity of its producer.

1.1 SUMMARY

Mobile phones are used for capturing and distributing (whatsapp, facebook) images for professional as well as personal uses. So, at some point we need to prove the ownership of the image .It is a kind of copyright that the image was captured by a particular mobile phone camera. Copyright is a form of intellectual property applicable to any expressible form of an idea or information that is substantive and discrete. Copyright prevents others from taking others work for free. It also prevents people from altering the work without permission. The problem arises when you have to prove that you are the owner of the image. Proposing a technique which embeds personal mobile phone numbers along with IMEI number inside the image using invisible watermarking technique. An invisible watermark is a pattern of bits inserted into a digital image file that identifies the file's copyright information.

1.2 PURPOSE

The application is for embedding and extracting the digital watermark which shows the ownership of the image.

It maintains two levels of users

- Developer level
- User level

The software includes

- Camera for taking picture and saving that image.
- Provides digital watermark to be automatically embedded in the image.
- Provides watermark to be revealed.
- All the information about application.

1.3 SCOPE

The System is designed for the authentication of image captured by mobile phone camera and it is only made for android devices. It includes the features that can embed digital watermark to the image captured by mobile phone camera and extract the watermark from that captured (watermarked) image. This application cannot be used on windows, IOS or any other OS.

1.4 TECHNOLOGY REVIEW

From Cupcake to KitKat A brief history of Google's Android Operating System. Since the introduction of Android in 2007, Google's flagship open-source OS for mobiles, the dynamics of the mobile phone industry have changed completely. A new mobile ecosystem, consisting of Android apps and mobile hardware has evolved at a rapid pace, challenging traditional market leaders like Apple, Nokia and Blackberry. If you ever wondered how this game-changing mobile OS has evolved over the years, then please read on. However, bear in mind that Google has an odd, albeit humorous, way of codenaming its Android OS products after mouthwatering desserts, so if you have a sweet tooth, I suggest that you find something sugary to enjoy while reading this post!

Evolution of the Android OS:

Google entered the mobile industry after acquiring Android Inc. in 2005, setting off rumors that it was planning on entering the mobile space. The rumors turned out to be true, as Google announced in November 2007 that it indeed was working on an open-source mobile OS, named Android, based on a Linux kernel. This new OS was to be used by members of the' Open Handset Alliance', a consortium of 65 companies involved in the mobile space who are advocates of open source standards for the mobile industry. Since then, mobile devices running on Android have gained rapid popularity among consumers, with Android OS currently dominating the smartphone market. Each major release of the Google Android OS is named after a sugary treat, in alphabetical order. Here is a quick flashback of all the different versions, along with the associated sugary delights:

Fig 1.1 : Android History

Android 1.0 and 1.1

The first version was released in September 2008, along with its launch device, 'HTC Hero'. Both the OS and the HTC device received favourable reviews. The dream of an open-source mobile ecosystem finally became a reality! Version 1.1 (released in February 2009) just came with a few updates and tweaks, with no major changes. At this stage, Google had not started naming its Android releases after delicacies, so these updates had no names assigned.

Android 1.5 Cupcake

Released in April 2009, this was the first major Android revision to get an official name by Google, heralding the start of the "dessert series" naming convention. With Cupcake, features like video uploading, text prediction and wireless music streaming became available.

Android 1.6 Donut

Released in September 2009, Donut came with major updates, the highlight of which was Google Maps. It also fixed OS reboot errors and enhanced the photo and video capabilities.

Android 2.0/2.1 Éclair

Version 2.0 was released in December 2009, followed by 2.1 in January 2010. These are considered to be a single release by most people, allowing added capabilities for Bluetooth, multi-touch support and live wallpapers, among other features.

Android 2.2 Froyo

Short for “Frozen Yoghurt”, this version was released in May 2010. It allowed for improved OS speed, supported hi-definition screen resolutions and Adobe Flash 10.1, enabling users to stream videos via their mobile browsers. Added support for Wi-Fi hotspot connectivity also became available.

Android 2.3 Gingerbread

This version made its debut in December 2010. The key feature that became available with this release was the much hyped ‘Near Field Communications’ (NFC) capability, allowing users to perform tasks such as mobile payments and data exchange through swiping their mobile phones over a tag. It also added support for more than one camera and other sensors.

Android 3.0/3.1/3.2 Honeycomb

Released first in February 2011 and followed rapidly by the 3.1 and 3.2 revisions during the same year, this incremental release added several new features. This version was optimized for tablets and provided developers with more control over UI. It also allowed users the capability to load media files directly from an SD card.

Android 4.0 Ice Cream Sandwich

Released in October 2011, this was a major overhaul to the Android UI, allowing enhanced contact menus, improved keyboard layouts and NFC capabilities. Since

Honeycomb was optimized just for tablets, with most phones still running the 2.x Android versions, the Ice Cream Sandwich release strove for a unified platform that was optimized to run on both tablets and phones.

Android 4.1.x Jelly Bean

The latest major Android update was released in July, 2012. Jelly Bean further polished the Android UI, and also refined the software, enabling Android devices to run faster and also making them even more user-friendly than before. The 4.x updates have allowed developers to create quality apps over Android, cementing its place as the operating system of choice among users.

Android 4.4 Kitket

Google announced Android 4.4 KitKat on September 3, 2013. Although initially under the "Key Lime Pie" ("KLP") codename, the name was changed because "very few people actually know the taste of a key lime pie." Some technology bloggers also expected the "Key Lime Pie" release to be Android 5. KitKat debuted on Google's Nexus 5 on October 31, 2013, and has been optimised to run on a greater range of devices than earlier Android versions, having 512 MB of RAM as a recommended minimum; those improvements were known as "Project Svelte" internally at Google. The required minimum amount of RAM available to Android is 340 MB, and all devices with less than 512 MB of RAM must report themselves as "low RAM" devices.

Overview of Android OS:

What is Android?

Android is a software stack for mobile devices that includes an operating system, middleware and key applications. The Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language.

Features

- ✓ **Application framework** enabling reuse and replacement of components
- ✓ **Dalvik virtual machine** optimized for mobile devices
- ✓ **Integrated browser** based on the open source Web Kit engine

- ✓ **Optimized graphics** powered by a custom 2D graphics library; 3D graphics based on the OpenGL ES 1.0 specification (hardware acceleration optional)
- ✓ **SQLite** for structured data storage
- ✓ **Media support** for common audio, video, and still image formats (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
- ✓ **GSM Telephony** (hardware dependent)
- ✓ **Bluetooth, EDGE, 3G, and WiFi** (hardware dependent)
- ✓ **Camera, GPS, compass, and accelerometer** (hardware dependent)
- ✓ **Rich development environment** including a device emulator, tools for debugging, memory and performance profiling, and a plug-in for the Eclipse IDE.

1.5 LITERATURE REVIEW

❖ What is Digital Watermark?

In general, a digital watermark is a code that is embedded inside an image. It acts as a digital signature, giving the image a sense of ownership or authenticity. Ideal properties of a digital watermark have been stated in many articles and papers.

These properties include:

1) A digital watermark should be perceptually invisible to prevent obstruction of the original image.

2) A digital watermark should be statistically invisible so it cannot be detected or erased.

3) Watermark extraction should be fairly simple. Otherwise, the detection process requires too much time or computation.

4) Watermark detection should be accurate. False positives, the detection of a unmarked image, and false negatives, the non-detection of a marked image, should be few.

5) Numerous watermarks can be produced. Otherwise, only a limited number of images may be marked.

6) Watermarks should be robust to filtering, additive noise, compression, and other forms of image manipulation

7) The watermark should be able to determine the true owner of the image.

Digital watermarking has been investigated deeply for its technical and commercial feasibility in all media types like, digital photographic image audio, printed materials or document images and video. It is a proven method for reducing content piracy and improving the ability to identify, tract and manage digital media. It is widely used in applications like rights management (Zhang, 2009), remote triggering, filtering/classification and e-commerce. It is a technique that is used to balance the need for content security with best possible consumer experience to enable media and entertainment industries to adapt the advanced facilities of the modern digital revolution while reducing the threat of content theft. In the present scenario, systems that have the ability to protect content (i) accurately (ii) rapidly (iii) reliably (iv) without invading privacy rights (v) cost effectively (vi) in a user-friendly manner and (vii) without drastic changes to the existing infrastructures, are highly desired. As commercial incentives increase, many new technologies for person authentication, information hiding and copyright protection are being developed, each with its own strengths and weaknesses and a potential function in commercial market. This chapter reviews some watermarking techniques.

Digital watermarking technology started as early as 1282 in Italy, where paper watermarks were used to indicate the paper brand and the mill that produced it. After this invention, the method quickly spread over Italy and then over Europe. Although originally intended for paper brand and mill identification, the technique was later enhanced to include paper format, quality and strength. They were also used to date and authenticate paper. During 18th century, this technique was first used for installing ant counterfeiting measures on money and other documents. They are still widely used as security features in currency today. An example is shown in Figure 2.1. These techniques were called Watermarking only during the end of 18th century. The first watermark, that is the base of today’s technology, is the patent filed in 1954 by Emil Hembrooke for identifying musical works.

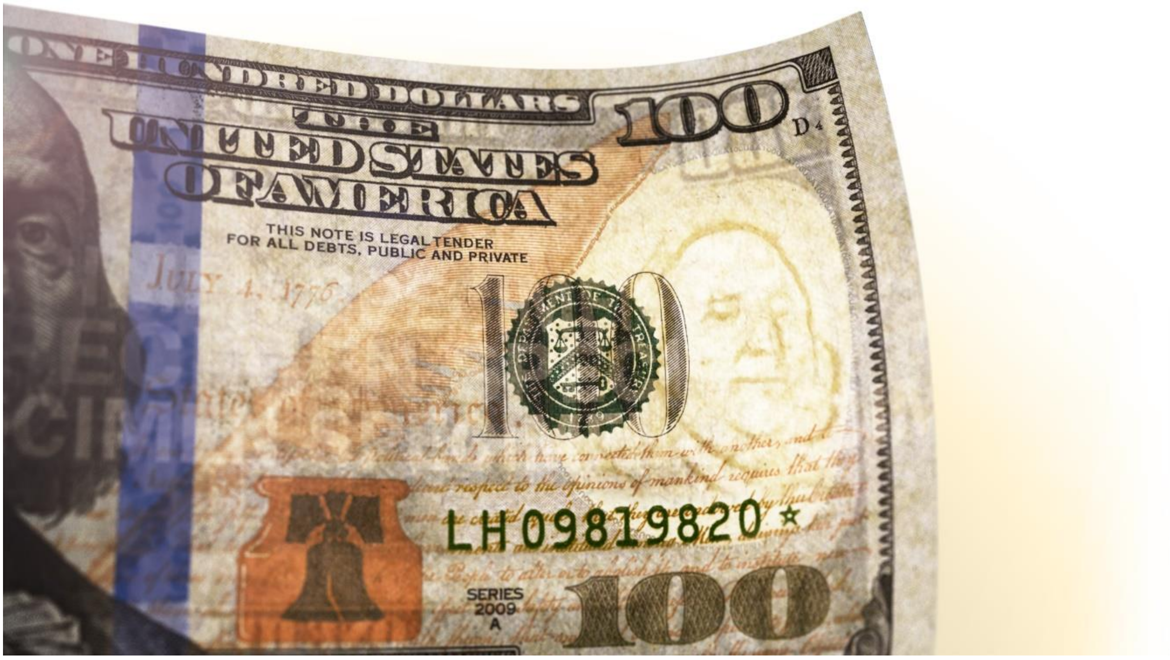


Fig 1.2: Old Watermarking System

The term "digital watermarking" came into existence only after 1988 and was coined by Komatsu and Tominaga (1988). Since then, there has been a huge interest in the field of digital watermarking and several different techniques have been proposed. Even though watermarks can be included with any digital content, this research focuses on image watermarking and the following sections review only those implementations that are related to this field.

Over the past few years, there has been tremendous growth in computer networks and more specifically the World Wide Web. This phenomenon, coupled with the exponential increase of computer performance, has facilitated the distribution of multimedia data such as images. Publishers, artists, and photographers, however, may be unwilling to distribute pictures over the Internet due to a lack of security, images can be easily duplicated and distributed without the owner's consent. Digital watermarks have been proposed as a way to tackle this tough issue. This digital signature could discourage copyright violation, and may help determine the authenticity and ownership of an image.

Digital watermark is a code that is embedded inside an image. It acts as a digital signature, giving the image a sense of ownership or authenticity. Watermarking is very similar to steganography in a number of respects. Both seek to embed information inside a cover message with little to no degradation of the cover-object. Watermarking however adds the additional requirement of robustness. An ideal stenographic system would embed a large amount of information, perfectly securely with no visible degradation to the cover object. Aniela watermarking system however would embed an amount of information that could not be removed or altered without making the cover object entirely unusable. We can also define a watermark as the digital data embedded in multimedia objects such that the watermark can be detected or extracted at later times in order to make an assertion about the object. The main purpose of digital watermarking is to embed information imperceptibly and robustly in the host data. Typically the watermark contains information about the origin, ownership, destination, copy control, transaction etc.

❖ Digital Image Watermarking Classification:

Some of the important types of watermarking based on different watermarks are given below:

Visible watermarks

Visible watermarks are an extension of the concept of logos. Such watermarks are applicable to images only. These logos are inlaid into the image but they are transparent. Such watermarks cannot be removed by cropping the center part of the image.

Invisible watermark

Invisible watermark is hidden in the content. It can be detected by an authorized agency only. Such watermarks are used for content and author authentication and for detecting unauthorized copier.

Fragile watermark

Fragile watermark are also known as tamper-proof watermarks. Such watermark are destroyed by data manipulation or in other words it is a watermarks designed to be destroyed by any form of copying or encoding other than a bit-for-bit digital copy. Absence of the watermark indicates that a copy has been made.

❖ Classification Of Image Watermarking Techniques:

The frequency sensitivity refers to the eye’s response to spatial, spectral, or time frequency changes. Spatial frequencies are perceived as patterns or textures, and spatial frequency sensitivity is usually described as the eye’s sensitivity to luminance changes [10]. It has been shown that an eye is the most sensitive to luminance changes in the mid-range spatial

frequencies, and that sensitivity decreases at lower and higher spatial frequencies. Digital image watermarking schemes mainly fall into two broad categories:
(i)Spatial-domain techniques..(ii)Frequency-domain techniques.

Spatial Domain Techniques

Spatial watermarking can also be applied using color separation. In this way, the watermark appears in only one of the color bands. This renders the watermark visibly subtle such that it is difficult to detect under regular viewing. However, the mark appears immediately when the colors are separated for printing. This renders the document useless for the printer; the watermark can be removed from the color band. This approach is used commercially for journalists to inspect digital pictures from a photo-stockhouse before buying unmarked versions.

Least Significant Bit(LSB)

The earliest work of digital image watermarking schemes embeds watermarks in the LSB of the pixels. Given an image with pixels, and each pixel being represented by an 8-bit sequence, the watermarks are embedded in the last (i.e., least significant) bit, of selected pixels of the image. This method is easy to implement and does not generate serious distortion to the image; however, it is not very robust against attacks. For instance, an attacker could simply randomize all LSBs, which effectively destroys the hidden information

SSM Modulation Based Technique

Spread-spectrum techniques are methods in which energy generated at one or more discrete frequencies is deliberately spread or distributed in time. This is done for a variety of reasons, including the establishment of secure communications, increasing resistance to natural interference and jamming, and to prevent detection. When applied to the context of image watermarking, SSM based watermarking algorithms embed information by linearly combining the host image with a small pseudo noise signal that is modulated by the embedded watermark.

Frequency Domain Techniques

Compared to spatial-domain methods, frequency-domain methods are more widely applied. The aim is to embed the watermarks in the spectral coefficients of the image. The most commonly used transforms are the Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT), Discrete Wavelet Transform (DWT), The reason for watermarking in the frequency domain is that the characteristics of the human visual system (HVS) are better captured by the spectral coefficients. For example, the HVS is more sensitive to low-frequency coefficients, and less sensitive to high-frequency coefficients. In other words, low-frequency coefficients are perceptually significant, which means alterations to those components might cause distortion to the original image. On the other hand, high-frequency coefficients are considered insignificant; thus, processing techniques, such as compression, tend to remove high-frequency coefficients aggressively. To obtain a balance between imperceptibility and robustness, most algorithms embed watermarks in the midrange frequencies.

Discrete Cosine Transformation (DCT)

DCT like a Fourier Transform, it represents data in terms of frequency space rather than an amplitude space. This is useful because that corresponds more to the way humans perceive light, so that the part that are not perceived can be identified and thrown away. DCT based watermarking techniques are robust compared to spatial domain techniques. Such algorithms are robust against simple image processing operations like low pass filtering, brightness and contrast adjustment, blurring etc. However, they are difficult to implement and are computationally more expensive. At the same time they are weak against geometric attacks like rotation, scaling, cropping etc. DCT domain watermarking can be classified into Global DCT watermarking and Block based DCT watermarking. Embedding in the perceptually significant portion of the image has its own advantages because most compression schemes remove the perceptually insignificant portion of the image.

Discrete Wavelet Transformation (DWT)

The Discrete Wavelet Transform (DWT) is currently used in a wide variety of signal processing applications, such as in audio and video compression, removal of noise in audio, and the simulation of wireless antenna distribution. Wavelets have their energy concentrated in time and are well suited for the analysis of transient, time-varying signals. Since most of the real life signals encountered are time varying in nature, the Wavelet Transform suits many applications very well. One of the main challenges of the watermarking problem is to achieve a better tradeoff between robustness and perceptivity.

Robustness can be achieved by increasing the strength of the embedded watermark, but the visible distortion would be increased as well. However, DWT is much preferred because it provides both a simultaneous spatial localization and a frequency spread of the watermark within the host image. The basic idea of discrete wavelet transform in image process is to multi-differentiated decompose the image into sub-image of different spatial domain and independent frequencies.

❖ Characteristics of Watermarking

There are many characteristics that watermarking hold are as follows:

Invisibility

An embedded watermark is not visible. Invisible watermark is hidden in the content. It can be detected by an authorized agency only. Such watermarks are used for content or author authentication and for detecting unauthorized copier.

Robustness

Piracy attacks or image processing should not affect the embedded watermark. Even if the visible watermark is removed (by an attack), there is the invisible one as the backup. The visible watermark is inserted into the original image while the invisible watermark is added to it. Therefore, it is a watermark within a watermark creating a dual watermarked image. This is another method of developing robust watermarking techniques. For robustness we can also add watermark at more than one position in the image, if one or two are removed then the other is there.

Security

A watermark should be secret and must be undetectable by an unauthorized user in general. A watermark should only be accessible by authorized parties. This requirement is regarded as a security and the watermark is usually achieved by the use of cryptographic keys. As information security techniques, the details of a digital watermark algorithm must be published to everyone. A particular watermark signal is related with a special number used embedding and extracting. The special number is kept secretly and is used for confirming legal owners of digital products later. If we lay strong stress on robustness, and then invisibility may be weak. Therefore, developing robustness watermark with invisibility is an important issue.

CHAPTER 2 : PROJECT MANAGEMENT

2.1 PROJECT PLANNING AND SCHEDULING

- **Project Planning**

1. Gather the module definition.
2. checking the time schedule feasibility.
3. Requirement gathering for module.
4. Analysis on gathered requirement.
5. Designing.
6. Coding.
7. Testing.
8. Management

2.1.1 PROJECT DEVELOPMENT APPROACH AND JUSTIFICATION

How to choose a particular Software Life Cycle Model:
Consider certain factors while choosing Software Life Cycle Model:

1. Continuing change: A large software system undergoes continuing change or becomes progressively less useful.
2. Increasing complexity: As software system evolves, its complexity increases unless work is done to maintain or reduce it.
3. Fundamental law of program evolution: Program evolution, the programming process, and global measures of project and system attributes are statistically self-regulating with determinable trends and invariance.

4. Invariant work rate: The rate of global activity in a large software project is statistically invariant.
5. Incremental growth limit: during the active life of a large program, the volume of modifications made to successive releases is statistically invariant.

❖ Justification of choosing Prototype Model:

Prototype model for the following reasons:

- I can repeat any process anytime again so as to refine the product features. And since I'm making the project for the first time, so it is obvious that I will make some mistakes.
- In our system, the requirement can't be freezing the needs. So, only the prototype model is appropriate.
- The feedback paths and communication with the customers allow for correction of the errors committed during a phase, as and when these are detected in a later phase.
- To reduces risk of incorrect user requirements only the prototype model is appropriate.
- Prototype model is good where requirements are changing/uncommitted in the system.
- Prototype model provides regular visible progress aids management.
- Prototype model supports early product marketing.

❖ The Prototype Model

- Prototyping is a technique that provides a reduced functionality or limited performance version of the eventual software to be delivered to the user in the early stages of the software development process. What is done is that before proceeding with design and coding, a throwaway prototype is built to give user a feel of the system.

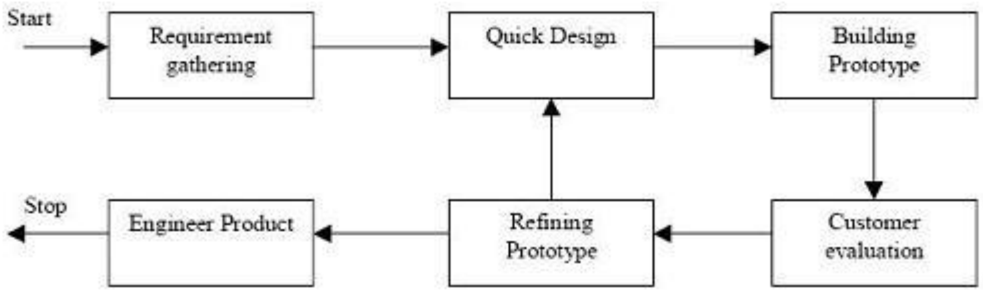


Fig 2.1 Prototype Model

The original purpose of a prototype is to allow users of the software to evaluate developers' proposals for the design of the eventual product by actually trying them out, rather than having to interpret and evaluate the design based on descriptions. Prototyping can also be used by end users to describe and prove requirements that developers have not considered, and that can be a key factor in the commercial relationship between developers and their clients Interaction design in particular makes heavy use of prototyping with that goal.

❖ **Summary:**

In this approach the prototype is constructed with the idea that it will be discarded and the final system will be built from scratch. The steps in this approach are:

- Write preliminary requirements
- Design the prototype
- User experiences/uses the prototype, specifies new requirements
- Repeat if necessary
- Write the final requirements
- Develop the real products

❖ **Advantages of Prototyping:**

The advantage of the prototyping model are as under:

- Users are actively involved in the development
- It provides a better system to users, as users have natural tendency to change their mind in specifying requirements and this method of developing systems supports this user tendency.
- Since in this methodology a working model of the system is provided, the users get a better understanding of the system being developed.

- Errors can be detected much earlier as the system is mode side by side.
- Quicker user feedback is available leading to better solutions.

❖ **Disadvantages Of Prototyping:**

- Leads to implementing and then repairing way of building systems.
- Practically, this methodology may increase the complexity of the system as scope of the system may expand beyond original plans.

I chose Waterfall model for the following reasons:

- In waterfall model, the flow is unidirectional, so there are no chances of going back to a particular process once it is completed.
- In this model, difficult to define all requirement in beginning.
- It needs heavy documentation.
- Small error may cause serious errors.·

Not, Incremental model for the following reasons:

- Each phase of an iteration is rigid and do not overlap each other.
- Sometime it is not possible to divide a problem.
- Less security to developers.

Not, Spiral model for the following reasons:

- It becomes too complex to implement.
- It is based on customer communication. If the communication is not proper then software product that gets developed will not be up to the mark.
- Difficult to have project estimation.

2.1.2 PROJECT PLAN INCLUDING MILESTONES, DELIVERABLES, ROLES, RESPONSIBLITIES AND DEPENDENCIES.

Feasibility analysis phase: 1 week

Image Watermarking module is feasible and can be implemented with help of above phases.

Requirement analysis and specification phase: 1 week

All the needed module is analyzed and tested.

Designing Phase : Approximately 3 week

Software design is a process by which the software requirements are translated into a representation of software components, interfaces and data necessary for the implementation phase.

Coding Phase : Approximately 5 week

- Firstly GUI related coding was done, followed by fetching, scanning of SMSs and categorization programming was done and after that note module was implemented.

Testing Phase : Approximately 1 week

- Test cases were designed and all modules are tested successfully.

2.2 RISK MANAGEMENT

2.2.1 RISK IDENTIFICATION

The objectives of risk identification are to identify and categorize risks that could affect the project and document these risks. The outcome of risk identification is a list of risks. What is done with the list of risks depends on the nature of the risks and the project. On noncomplex, low-cost projects with little uncertainty (few risks), the risks may be kept simply as a list of red flag items. On complex, high-cost projects that are by nature uncertain, the risks can feed the rigorous process of assessment, analysis, mitigation and planning, allocation, and monitoring and updating described in this document.

2.2.2 RISK ANALYSIS

Risk analysis should be performed as part of the risk management process for the project. The data of which would be based on risk discussion to identify potential issues and risks ahead of time before these were to pose cost and/ or schedule negative impacts .

In quantitative risk analysis, an attempt is made to numerically determine the probabilities of various adverse events and the likely extent of the losses if a particular event takes place.

Qualitative risk analysis, which is used more often, does not involve numerical probabilities or predictions of loss. Instead, the qualitative method involves defining the various threats, determining the extent of vulnerabilities and devising countermeasures should an attack occur.

2.2.3 RISK PLANNING

The risk planning contains an analysis of likely risks with both high and low impact, as well as mitigation strategies to help the project avoid being derailed should common problems arise. Risk planning should be periodically reviewed to avoid having the analysis become stale and not reflective of actual potential project risks.

Most critically, risk management plans include a risk strategy. Broadly, there are four potential strategies, with numerous variations. Projects may choose to:

- Avoid risk — Change plans to circumvent the problem;
- Control risk; — Reduces impact or likelihood (or both) through intermediate steps;
- Accept risk — Take the chance of negative impact (or auto-insurance), eventually budget the cost (e.g. via a contingency budget line);
- Transfer risk — Outsource risk (or a portion of the risk - Share risk) to third party/ies that can manage the outcome. This is done e.g. financially through insurance contracts or hedging transactions, or operationally through outsourcing an activity.

CHAPTER 3 : SYSTEM REQUIREMENTS STUDY

3.1 USER CHARACTERISTICS

Any user who uses compatible android version in their particular smart phones can deal/use this application. Data security and data transfer in the main functionality provided by this application. The user of this application needs to be conscious about the data security and its importance. However any user who is aware of or has basic knowledge of mobile can use it. However this application can be independently be used for transferring the file from one machine to another machine.

3.2 HARDWARE AND SOFTWARE REQUIREMENTS

❖ **Software Tools:**

1. I will use Java for rendering the User Interface, for which, Android 4 and above with JVM and SDK is a must.

2. Eclipse 3.8.2 will be used for all programming practices.

3. SQLite3 database for storing and accessing the text formatted advertisements and other text formatted information.

5. I use java.lang Java library in our Application.

Operating System : Android 2.3 or Above.

Front End Tool : Eclipse

❖ **Hardware Tools:**

Since neither the database application nor the application have any designated hardware, it does not have any direct hardware interfaces.

600 MHz Mobile Processor. 160MB RAM.

50 MB of Phone Memory. Android Smart Phone.

CHAPTER 4 : SYSTEM ANALYSIS

Analysis focuses on creation of models. Analysis capture & scrutinize requirement by constructing models. They specify what must be done, not how it should be done. Analysis is a difficult task in its own right, & developers must fully understand the problem before addressing the complexities of design.

4.1 STUDY OF CURRENT SYSTEM

In the current system for embedding watermark observed that the watermark is embedded manually. And most of the applications are for adding the visible watermark. When the watermarked image falls in the hand of the hacker then the hacker can insert, delete or modify the content of the original watermark.

4.2 PROBLEMS AND WEAKNESSES OF CURRENT SYSTEM

You may find some disadvantages associated with digital watermarking. The first disadvantage is the fact that,

1) Your image may look less appealing to others, with a watermark.
2) Adding watermark may be manually which is time consuming.

4.3 REQUIREMENTS OF NEW SYSTEM

Functional Requirements:

- It should be capable of identifying the authorised and unauthorised user.
- It should be capable of embedding IMEI and MSISDN numbers into image file.
- On embedding the message the look of the image should not be destroyed.

- Reverse to embedding the watermark it should also retrieve the message from file as it was embedded.
- After embedding the watermark it should retrieve the watermark from the file in the same format in which the watermark was previously embedded.
- After capturing the image the image should be saved in the gallery.

Non-Functional Requirements:

- **Reliability**

The application has to be very reliable due to the importance of data and the damages incorrect or incomplete data can do.

Availability: The application is available 100% for the user and is used 24 hrs Day and days year. The system shall be operational 24 hours a day and 7days a week..

Mean Time to Repair (MTTR): Even if the application fails, the system will be recovered back up within an hour or less.

Access Reliability: The application shall provide 100% access reliability.

- **Maintainability**

The application should be designed to be easily maintainable and get the least complaints from users, along with minimum downtime.

- **Extensibility**

The application should be designed to be extensible to changes. Addition of new updates should be smooth and easy.

4.4 FEASIBILITY STUDY

Feasibility Study is the test of the system proposal according to its work ability, impact on the current system, ability to meet the needs of the current users and effective use of the resources. Its main objective is not to solve the problem, but to acquire its scope. It focuses on following:

- Meet user requirements.
- Best utilization of available resources.
- Develop a cost effective system.
- Develop a technically feasible system.

There are three aspects in the feasibility study:

- Technical Feasibility
- Economical Feasibility
- Behavioral Feasibility

Economically Feasibility:

- The system being developed is economic with respect to company's point of view. It is cost effective in the sense that has eliminated the paper work completely. The system is also time effective because the calculations tasks, tracking tasks and storing tasks are fully automated which are made as per the user requirement. Minimum errors and are highly accurate as the data is required.
- Internet/Wi-Fi connectivity is needed in Android Device to use the application.

Technical Feasibility:

- The technical requirement for the system is economic and it does use SIP protocol supported Android Devices Hardware and Android OS. It also requires company's Proxy Server (Deploying Web Service).

Behavioral Feasibility:

- The system working is quite easy to use and learn due to its simple user interface. User requires no special training for operating the system.

4.5 FUNCTION OF SYSTEM

4.5.1 USE CASES EVENT TRACE OR SCENARIO

Use cases are best discovered by examining what the actor needs and defining what the actor will be able to do with the system; this helps ensure that the system will be what the user expects.

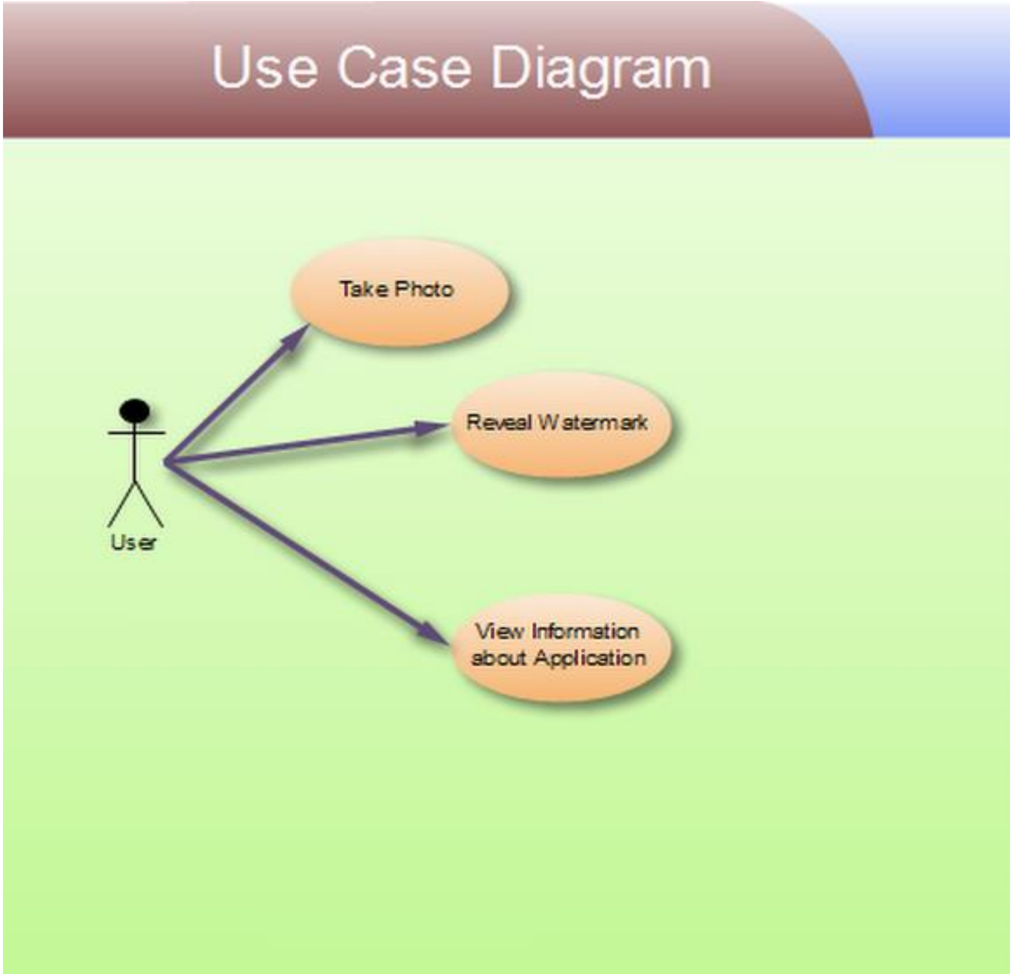


Fig 4.1 : Use Case Diagram

4.5.2 ACTIVITY DIAGRAM

Activity diagram for user

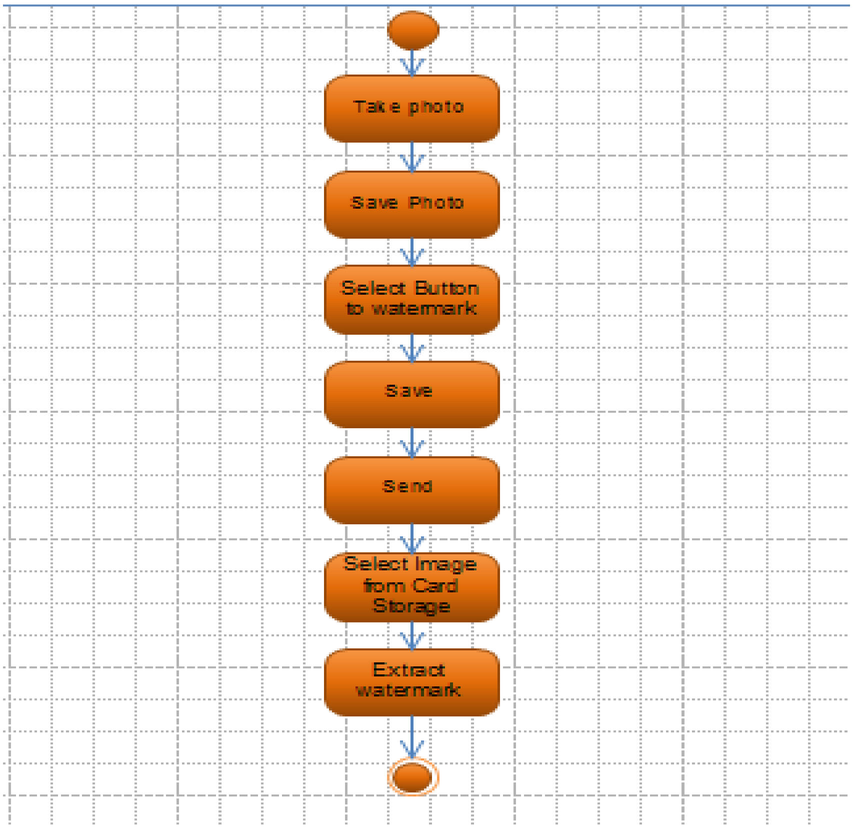


Fig 4.2 : Activity Diagram For User

Activity diagram for developer:

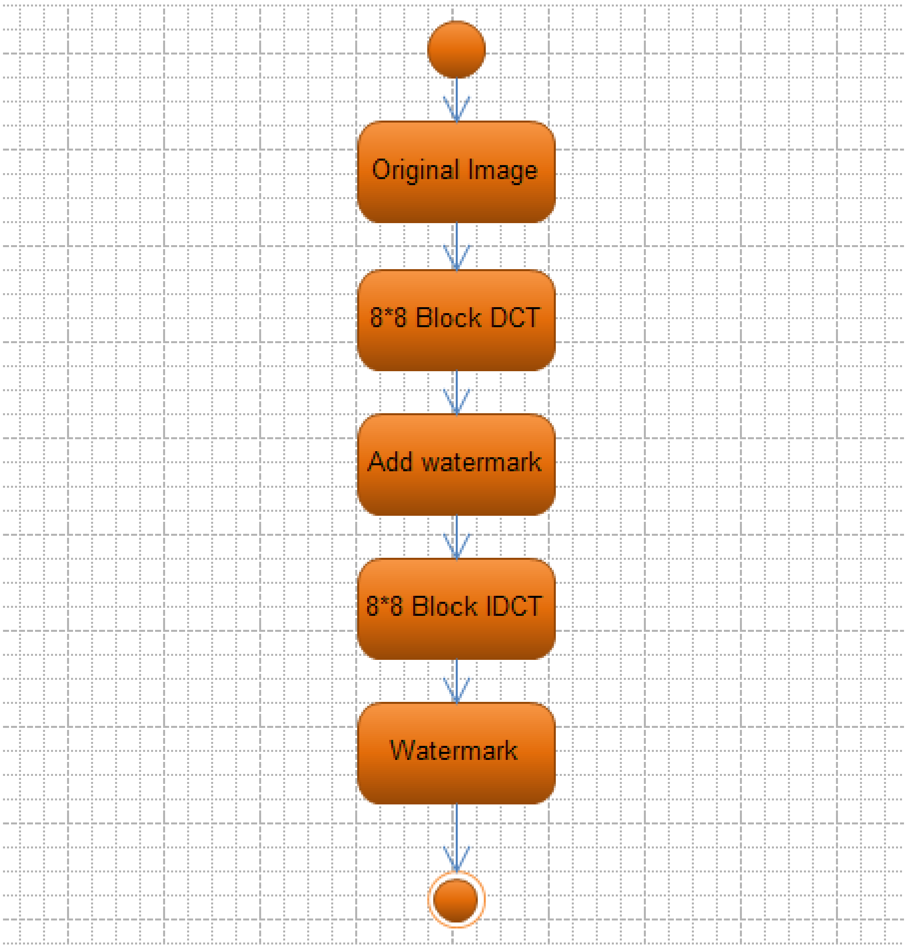


Fig 4.3 : Activity Diagram for Developer

CHAPTER 5 : MAIN MODULES OF THE NEW SYSTEM

There are two main modules of this system and they are:

- Embedding the watermark
- Extracting the watermark.

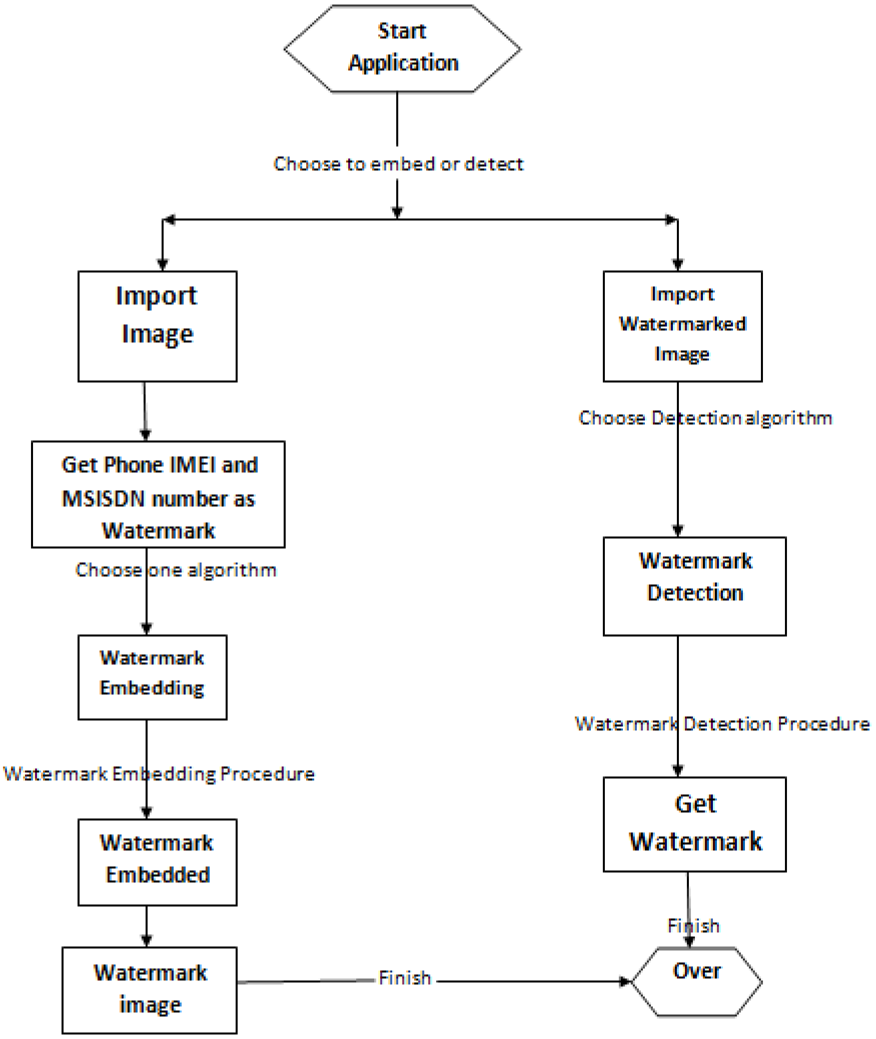


Fig 5.1 : Whole Application Flow

This diagram shows the flow of the whole application. There are two main modules namely Embedding Watermark and Extracting Watermark. The flow towards left shows the Embedding Watermark and the rest shows the Extracting Watermark.

5.1 EMBEDDING WATERMARK

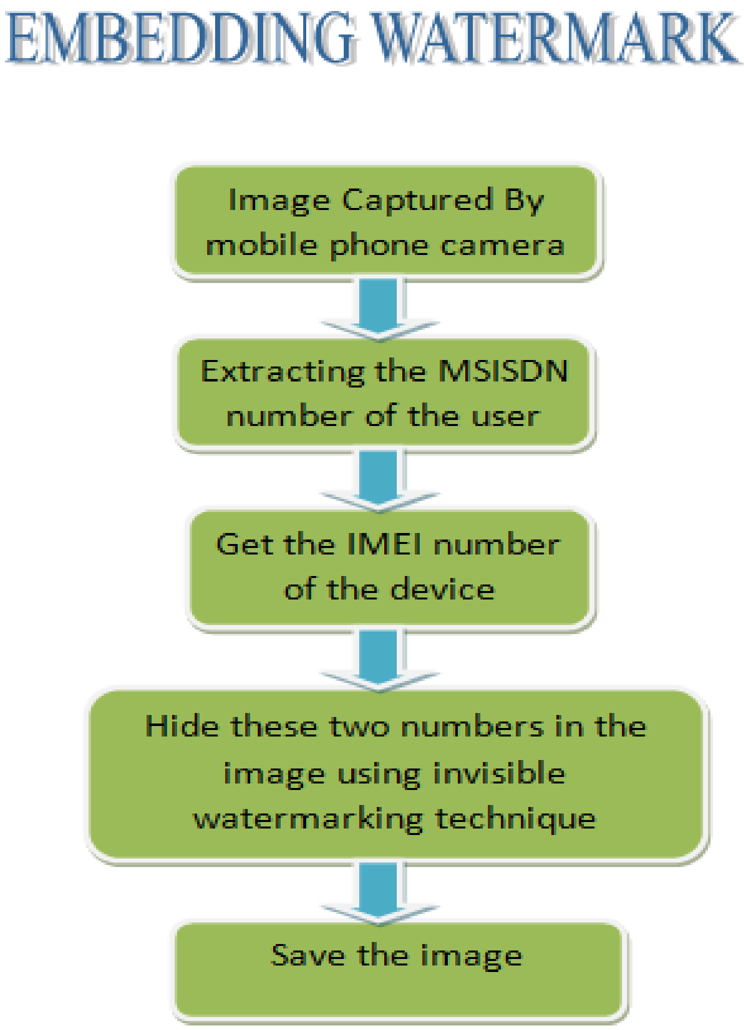

EMBEDDING WATERMARK
Image Captured By mobile phone camera
Extracting the MSISDN number of the user
Get the IMEI number of the device
Hide these two numbers in the image using invisible watermarking technique
Save the image

Fig 5.2 : Flow Of Embedding Watermark

In recent years, Smart phone has become a popular consumer electronics product. Young people like to use that to record their daily lives, moreover, they will share these photos and information to others. However, the photos may be used without consent after they are uploaded to Internet. To avoid this problem, one can embed visible and invisible watermarks into images. However, an additional process for embedding watermarks should be performed before an image is uploaded. So, propose a copyright embedding system for Android platform. Using this system pre-specified copyright information is automatically embedded into pictures with digital watermark technology when these pictures are taken.

In addition, original images (i.e., images without watermarks) can be preserved selectively. This system has following features:

(1) Computational complexity of watermark embedding process is possibly reduced for handheld mobile system.

(2) The watermark can be extracted without the use of the original image.

(3) The watermark embedded into an image would not be removed by commonly used image processing operations.

(4) Embedding copyright information, resizing the images, and uploading images to Internet are automatically performed without manual intervention.

Therefore, this system is very suitable for the protection of the photographs taken by Android phones to prevent piratical behaviors.

5.2 EXTRACTION WATERMARK

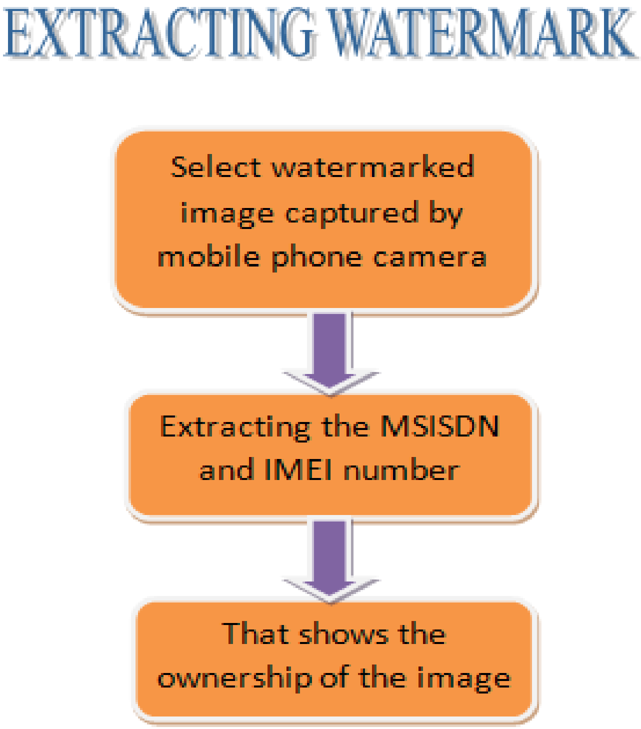

EXTRACTING WATERMARK
Select watermarked image captured by mobile phone camera
Extracting the MSISDN and IMEI number
That shows the ownership of the image

Fig 5.3 : Flow of Extracting Watermark

Whenever the is need to prove the ownership of the images captured by mobile phone camera then, by extracting the watermarked image we can get the data embedded in the image. In this application the IMEI and MSISDN numbers are embedded inside the image using digital watermark. So we can get the ownership of the image by extracting the IMEI and MSISDN numbers from the image. By using IDCT algorithm, exactly reverse process is done as that of in DCT algorithm for watermarking.

CHAPTER-6: IMPLEMENTATION PLANNING AND DETAILS

6.1 IMPLEMENTATION ENVIRONMENT

- JAVA Programming Language.
- Ecllipse as a platform
- AVD-5554 as an Emulator
- Android Studio SDK

6.2 SECURITY FEATURES

The Security issues are less in the application because this application embeds the IMEI (International Mobile Station Equipment Identity) and MSISDN(Mobile Station

International Subscriber Directory Number) are embedded which respectively shows the device id and the personal mobile phone number as digital watermark.

6.3 CODING STANDARDS

Any good software development approach suggests to adhere to some well-defined standards or rules for coding. These rules are called coding standards.

❖ NAMING CONVENTIONS

Following are some commonly used naming conventions in coding.

- Packages name and variable name should be in lower case.
- Variable names must not begin with numbers.
- The type name should be noun and it should start with capital letter.
- Constants must be in upper case.
- Method name must be given in lower case.
- The variables with large scope must have long name. For example count_total, sum.
- Variable with short scope must have short name. For example i,j.
- The prefix is must be used for Boolean type of variable. For example isEmpty or isFull.

❖ FILES

Reader must get an idea about the purpose of the file by its name. In some programming languages like Java-

- The file extension must be java.
- The name of the file and the class defined in the file must have same name.
- Line length in the file must be limited to 80 characters.

❖ COMMENTING/LAYOUT

Comments are non executable part of the code. But it is very important because it enhances the readability of the code. The purpose of the code is to explain the logic of the program.

- Single line comments must be given by //.
- For the names of the variables comments must be given.
- A block of comment must be enclosed within /* and */.

❖ **STATEMENTS**

These are some guidelines about the declaration and executable statements.

- Declare some related variables on same line and unrelated variables on another line.
- Class variable should never be declared public.
- Make use of only loop control within the for loop.
- Avoid use of break and continue statements in the loop.
- Avoid complex conditional expressions. Make use of temporary variables instead.
- Avoid the use of do…while statement.

6.4 SAMPLE CODING

❖ **Using the following code we can capture the image**

```
public void onClick(View v) {
                Intent cameraIntent = new Intent(
          android.provider.MediaStore.ACTION_IMAGE_CAPTURE);
                startActivityForResult(cameraIntent, 3);}
```

❖ **After Capturing the image we will embed the IMEI and MSISDN numbers using the following line of code**

```
protected void onCreate(Bundle savedInstanceState) {
          super.onCreate(savedInstanceState);
```

```
                        setContentView(R.layout.activity_main);
                        tel = (TelephonyManager)
        getSystemService(Context.TELEPHONY_SERVICE);
                        tel= (TelephonyManager)
        getLine1Number(Context.TELEPHONY_SERVICE);

                imei = (TextView) findViewById(R.id.textView2);
                imei.setText(tel.getDeviceId().toString());
```

- **Embedding and Extracting are done using DCT and IDCT respectively.**

CHAPTRR-7: TESTING

7.1 TESTING PLAN

A test plan is a general document for the entire project that defines the scope, approach to be taken, and the schedule of testing as well as identifies the test items for the entire testing process and the personnel responsible for the different activities of testing.

The test planning can be done well before the actual testing commences and can be done in parallel in the design and coding phase.

The input for the test plan is:

1) Project Plan
2) Requirement Document and
3) System Design Document.

The project plan is needed to make sure that the test plan can be consistent with the overall plan for the project and the testing schedule matches that of the project plan. The requirement documents and the design document are the basic documents used for selecting the test unit and the deciding the approaches to be used during testing.

A test plan should contain following:

1) Test Unit Specification.

2) Features to be tested.

Test Unit

A test unit is a set of one or more modules, together which associated data, that are from single computer program and that are the object of testing. A test unit can occur at any level and can contain from a single module to entire system.

Features to be tested

All functional features specified in the requirement document will be tested. The Features to be tested are:

1) Embed watermark

2) Retrieve watermark

3) Send Watermarked image.

7.2 TESTING STRATEGY

The development process repeats this testing sub process a number of lines for the following phases.

- Unit Testing
- Integration Testing

Unit Testing tests a unit of code after coding of that unit is completed. Integration Testing tests whether the previous programs that make up a system, interface with each other as desired System testing ensures that the system meets its stated design specifications. Acceptance testing is testing by users to ascertain whether the system developed is a correct implementation of the software requirements specification.

Testing is carried out in such a hierarchical manner to that each component is correct and the assembly/combination of component is correct .Merely testing a whole system at end would most

likely throw up errors in component that would be very costly to trace and fix. Also, performed both Unit Testing and System to detect and fix errors. A brief of is given below

- **Unit Testing**

Objectives:

The objective of unit testing is to test a unit of code using the unit test specification,after coding is completed, since the testing will depend on the completeness and correctness of the test specification, it is

important to subject these to quality and verification reviews.

Input: Unit test Specifications.

- **Testing process**
  - Checking for availability of code walk – through reports which have documented the existence of and conformance to coding standards.
  - Review of test specification.
  - Verify the unit test specification conforms to the program specification.

7.3 TESTING METHODS

Black Box Testing

This method treats the coded module as a black box. The module runs with inputs that are likely to cause errors. Then the output is checked to see if any error occurred. This method cannot be used to test all errors, because some errors may depend on the code or algorithm used to implement the module.

Black Box Testing, also known as Behavioral Testing, is a software testing method in which the internal structure/design/implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional.

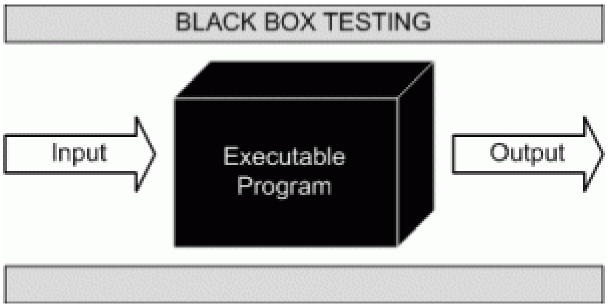


Fig 7.1 : Black Box Testing

This method is named so because the software program, in the eyes of the tester, is like a black box; inside which one cannot see.

7.4 TEST CASES

Test case 1: Embed Watermark

User can embed the message behind the image. For that user has to enter the Input file and the watermark which is to be embedded.

If the Embed process is successful then it will show message, Image Saved”. If at all the embed process do not complete successfully, then it will generate an alert.

Either of the fields should not be left blank. If any of the field is left blank then it will show an alert message.

Test case 2: Extract Watermark

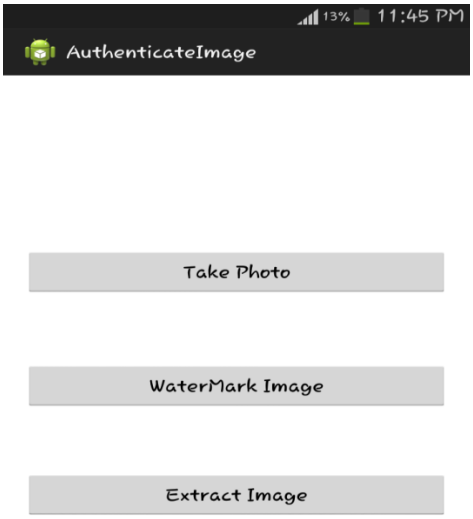
- User can retrieve watermark from the image. For that user has to enter the Input file in which the watermark has been embedded. Then user has to click on retrieve button to get the content.

Test case 3: Send Image

- User can send message to the desired destination using photo sharing application.
- If the user enters the wrong destination name or address then it will show alert message.

CAPTER-8 SCREEN SHOTS AND USER MANUAL

When user click on the application icon



When user click on take photo

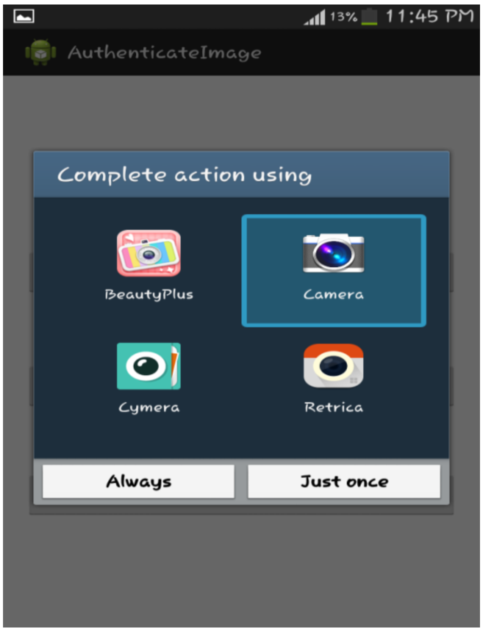


Fig 8.1 Main Page

When user selects the camera

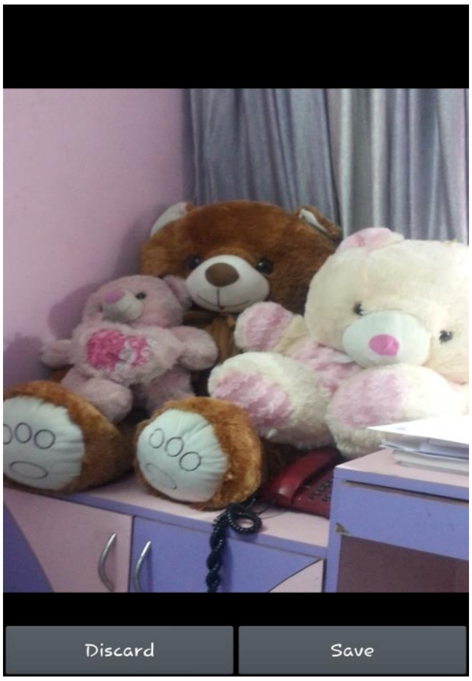


Fig 8.2 : popup

click on the watermark image

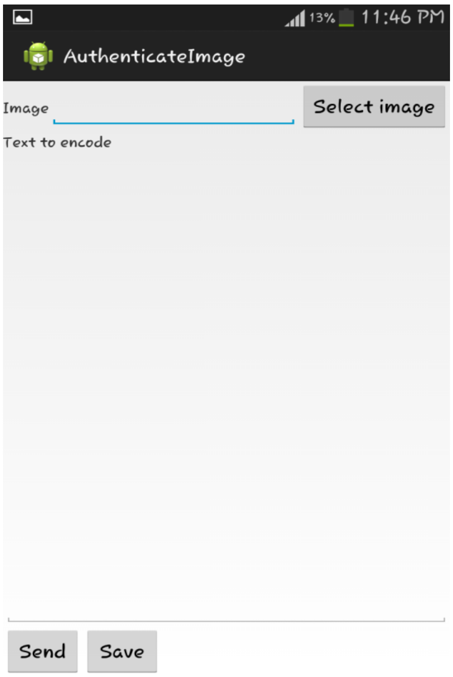


Fig 8.3 : Save Image

When user click on select image

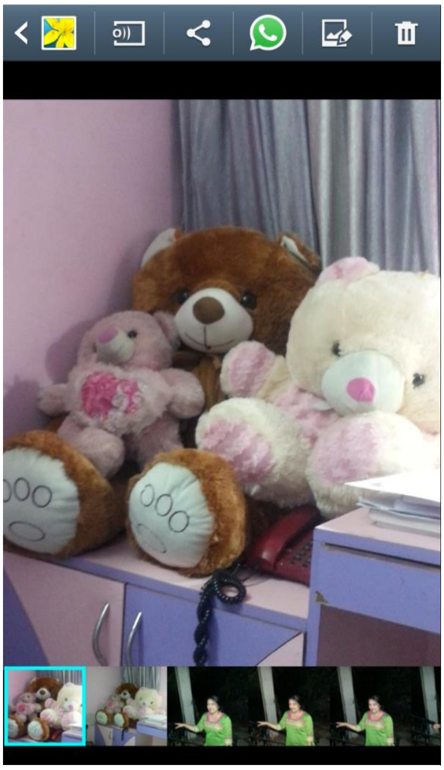


Fig 8.4 : Select image and text

When user selects the image

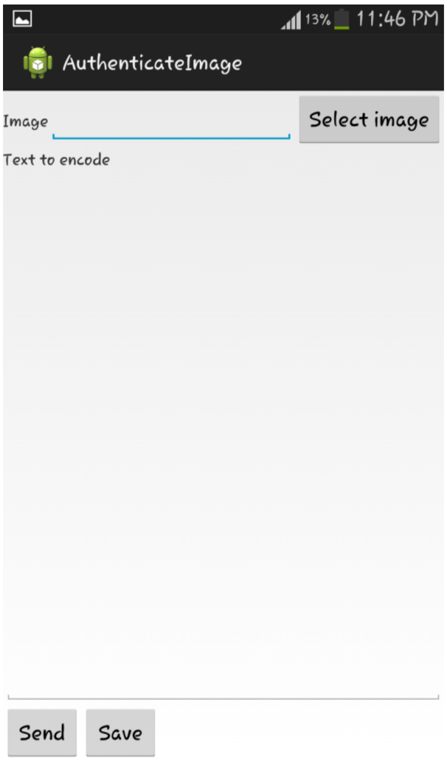
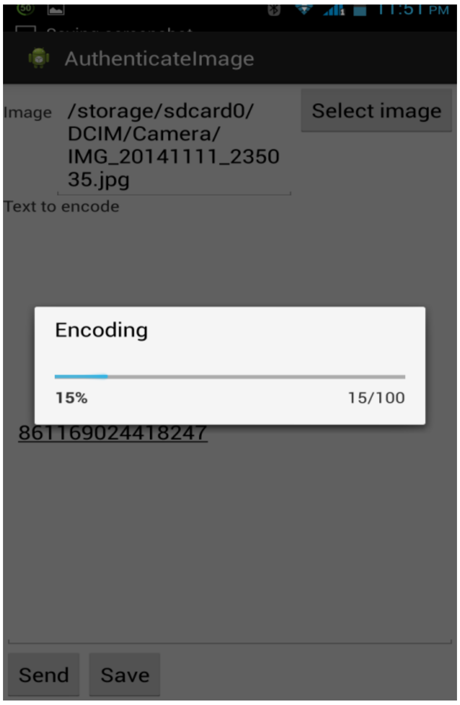


Fig 8.5 : Select image from gallay **Fig 8.6 : Select send to share**

Now select the image from the gallery and enter the data to be watermarked and save that image.



AuthenticateImage

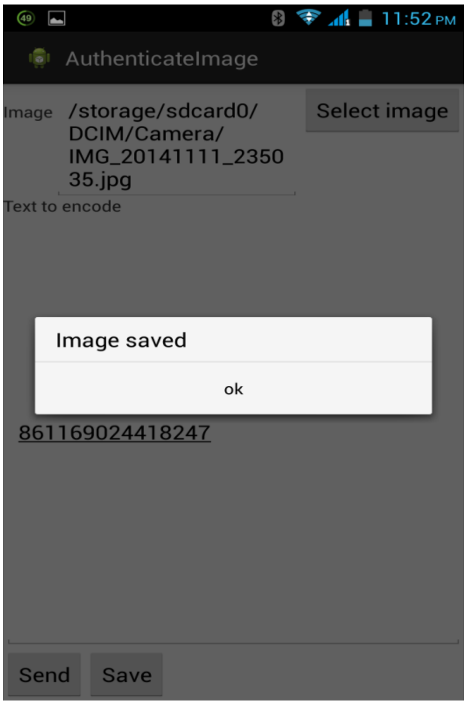
Image /storage/sdcard0/DCIM/Camera/IMG_20141111_235035.jpg

Text to encode

Encoding

15% 15/100

861169024418247



AuthenticateImage

Image /storage/sdcard0/DCIM/Camera/IMG_20141111_235035.jpg

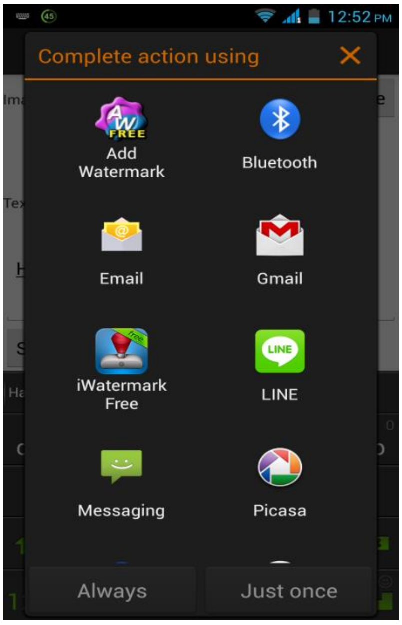
Text to encode

Image saved

861169024418247

Fig 8.7 : Progressbar **Fig 8.8 : Image Saved**

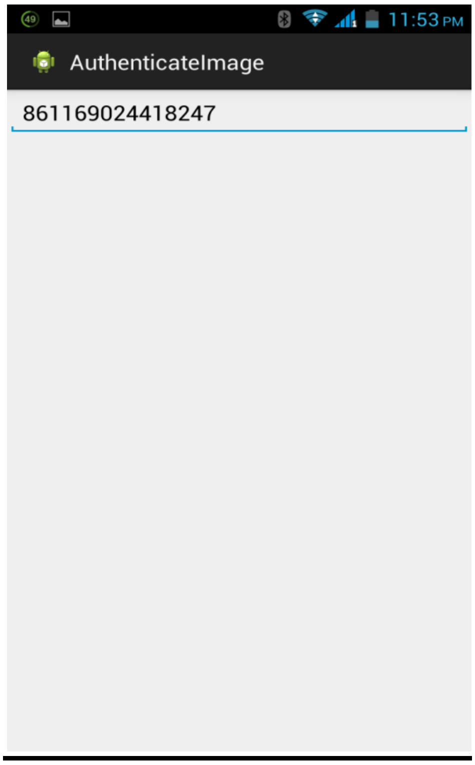
When selecting send button to share watermarked image



Complete action using

Fig 8.9 : Share image

Now select extract watermark button and select the watermarked image which was automatically saved in the SD card storage.



861169024418247

Fig 8.10 : Watermark

CHAPTER-9 LIMITATION AND FUTURE ENHANCEMENT

LIMITATION

1) The application cannot embed message in text file, audio file or video file. If it tries to embed message in text file, audio file or video file then it will show distorted file.

2) Only text file can be embedded in image. No other files can be embedded.

3) The message cannot be transferred to any other device or we can say any other OS.

4) This application is not designed for any other OS except Android.

5) If any application which does not support the image having large space then that watermarked image is of no use.

FUTURE ENHANCEMENT:

1) Security will not lose by any mean.

2) The application can embed watermark in text, audio or video file.

3) Not only text file but also any other file can be embedded in image, audio and video file.

4) The watermarked image can be transferred in intranet and in Internet environment.

5) The watermark can be embedded in one OS and it can be Extracted in another OS.

CHAPTER-10 CONCLUSION AND DISCUSSION

I embedded MSISDN (personal mobile phone number) along with IMEI number of the device inside the image using invisible watermarking technique. By extracting the watermark we will get the ownership of the image captured by mobile phone camera.

References

Setyawan, Iwan. "Watermarking Digital Image and Video Data. A State-of-the-Art Overview."*IEEE Signal Processing Magazine*, www.academia.edu/6337945/Watermarking_digital_image_and_video_data._A_state-of-the-art_overview.

[2]. A Fast and Robust Digital Watermark Detection Scheme for Cellular Phones, Takao Nakamura, Atsushi Katayama, Ryo Kitahara, and Kenji Nakazawa, NTT Cyber Space Laboratories Yokosuka-shi, 239-0847 Japan, 2006

[3]. Android System Smart-Phone Hardware Configuration. http://www.engadget.com/2011/04/15/htc-sensation-versus-therest-of-the-dual-core-world-smartphone/

[4]. Towards Robust and Hidden Image Copyright Labeling, E. Koch & J. Zhao, Fraunhofer Institute for Computer Graphics Wilhelminenstr. 7, 64283 Darmstadt, Germany, Proc. of 1995 IEEE Workshop on Nonlinear Signal and Image Processing (Neos Marmaras, Greece, June 20-22, 1995)

[5]. Literature Survey on Digital Image Watermarking Er-Hsien Fu EE381K-Multidimensional Signal Processing 8/19/98.

[6]. How to programmatically get the devices IMEI/ESN in Android? http://stackoverflow.com/questions/1972381/how-to-programmatically-get-the-devices-imei-esn-in-android

[7]. How to retrieve the Device Unique ID from android device http://developer.samsung.com/android/technical-docs/How-to-retrieve-the-Device-Unique-ID-from-android-device

[8]. Getting IMEI Number and other Details**.** http://www.learn-android-easily.com/2013/05/getting-imei-number-and-other-detail-html