

Location Privacy Preservation Using Transformed Features

Submitted in partial fulfillment of the required for the
award of

Bachelors of Computer Science

by

Om Jaya Prakaash.S

REG NO : (40290064)



DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF SCIENCE AND HUMANITIES

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

**Accredited with Grade "A" by NAAC | 12B Status by UGC |
Approved by AICTE**

JEPPIAR NAGAR, RAJIV GANDHI SALAI, CHENNAI 600 199

MAY 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC|12B Status by UGC| Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND SCHOOL OF
COMPUTING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **OM JAYA PRAKAASH S (40290064)** who carried out the project entitled "LOCATION PRIVACY PRESERVATION USING TRANSFORMED FEATURES" under our supervision from JAN 2023 TO MARCH 2023.

INTERNAL GUIDE

Dr.V.ULAGAMUTHALVI, M.E.,Ph.D.

DEAN AND HEAD OF THE DEPARTMENT

Dr.REKHA CHAKRAVARTHI M.E.,Ph.D

Submitted for Viva voce Examination held on 08.05.2023

Internal Examiner

External Examiner

DECLARATION

I am, **OM JAYA PRAKAASH (40290064)** here by declare that the Project Report entitled “**Location Privacy Preservation Using Transformed Features**” done by under the guidance of **Dr.V.ULAGAMUTHALVI, M.E.,Ph.D**, Department of Computer Science at **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY**, Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai-600 119 is submitted in partial fulfillment of the requirements for the award of Master of Science degree in Computer Science.

DATE:

PLACE :



SIGNATURE OF THE CANDITATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for

Their kind encouragement in doing this project and for completing it successfully .I am grateful to them.

I convey my thanks to **Dr. REKHA CHAKRAVARTHI M.E., Ph.D.Dean of Arts and Science** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sence of gratitude to my project guide **Dr.V.ULAGAMUTHALVI, M.E.,Ph.D**, for their valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff member of the Department of Computer Science school computing who were helpful in many ways for the completion of the project

OM JAYA PRAKASH

ABSTRACT

Privacy protection in Content Based Image Retrieval (CBIR) is a new research topic in cyber security and privacy. The privacy of the images is the main problem in the current world where some systems identifies the user current location from the images which they will be stored in the server. This process is called as Content Based Image Retrieval (CBIR) where the contents of the image are retrieved. The location of the user is protected by the encryption process. But, this implementation does not lead a good and reliable protection on the user spot identification. Since, the encryption is made only after the location is stored in the database. Here it is proposed a relevance encryption free privacy based system where the user spot is not known to the server and the traitor who hacks the server. The implementation get execute on two processes: (1) extraction of the features from the image of the user and (2) transformation of the features from the extracted features.

TABLE CONTENT

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	V
	LIST OF FIGURES	VIII
1	INTRODUCTION	1
	1.1 CYBER SECURITY INTERCONNECTED SYSTEM	2
	1.2 SCOPE OF PROJECT	3
	1.3 LITERATURE OF SURVEY	3
2	AIM AND SCOPE OF PRESENT INVESTIGATION	8
	2.1 AIM OF PROPOSED SYSTEM	8
	2.2 SCOPE OF PROPOSED SYSTEM	9
	2.3 EXISTING SYSTEM	10
3	SYSTEM CONFIGURATION	11
	3.1 SYSTEM ARCHITECTURE	11
	3.2 HARDWARE REQUIREMENT	11
	3.3 SOFTWARE REQUIREMENT	12
4	EXPRIMENTS, METHODS AND ALGORITHM	13
	4. 1 METHODOLOGY	13
	4.1.1 USER REGISTRATION AND LOGIN	13
	4.1.2 DATA FLOW DIAGRAM	13
	4.1.3 FEATURE EXTRACTION	14
	4.1.4 TRANSFORMED FEATURES	14
	4 .2 ALGORITHM USED	15
	4.2.1 SIFT TECHNIQUES	15
	4.2.1.1 GAUSSIAM PYRAMID	15

	4.2.1.2 EXPTEMA DETECTION	16
	4.2.1.3 CONVOLUTION NEURAL NETWORK (CNN)	17
5	RESULTS AND DISCUSSION	19
	5.1 DISCUSSION	19
	5.2 RESULT	20
	5.2.1 UNIT TESTING	20
	5.2.2 SYSTEM TESTING	20
	5.2.3 INTEGRATION TESTING	21
	5.3 TEST SCREENS	22
6	CONCLUSION	25
	6.1 APPENDIX	26
	6.2 SOURCE CODE	26
	6.3 SCREENSHOT	52
7	REFERENCE	59

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
3.1	System Architecture	11
4.1.1	Data Flow Diagram of User Login Credentials	13
4.2.1.3	Structure of The Artificial Neural Network	18

CHAPTER 1

INTRODUCTION

For content level access, very often database needs the query as a sample image. However, the image may contain private information and hence the user does not wish to reveal the image to the database. Private Content Based Image Retrieval (PCBIR) deals with retrieving similar images from an image database without revealing the content of the query image . not even to the database server. Private Content Based Image Retrieval (PCBIR) deals with the retrieval of similar content without revealing the content of the query image to the database. PCBIR is achieved by exchange of messages between the user and the database. These messages collectively help the user in inferring the required information from the database but prohibit the database from knowing the user's interest. In this sense, PCBIR is similar to private information retrieval (PIR) schemes that allow a user to obtain the data stored at a specific address in a database whilst keeping the database oblivious of the address. Since image retrieval is fundamentally a similarity search, PCBIR can be more efficiently solved (than PIR) as shown in Section 4. The lower bound on server side computation for private retrieval of arbitrary information is obviously $\Omega(n)$. This is because if the server does not read the entire database, something about the query is guaranteed to be revealed! However, in the case of images, concepts appear in the form of clusters. Consequently, it is enough for the server to just touch all the clusters for the query's privacy. Note that for optimizing the privacy, server should have the identical access patterns in all the clusters. The traditional CBIR research focuses on feature selection, efficient indexing as well as high performance in the form of recall and precision. PCBIR is concerned with efficient retrieval under the privacy constraint without trading the recall and precision. The approach may also be viewed as a secure computation of a similarity measure.

1.1 CYBER SECURITY INTER CONNECTED SYSTEM

Cyber security is primarily about people, processes, and technologies working together to encompass the full range of threat reduction, vulnerability reduction, deterrence, international engagement, incident response, resiliency, and recovery policies and activities, including computer network operations, information assurance, law enforcement, etc."

Cyber security is the protection of Internet-connected systems, including hardware, software, and data from cyber attacks. It is made up of two words one is cyber and other is security. Cyber is related to the technology which contains systems, network and programs or data. The security related to the protection of the systems security, network security and application and information security. It is the body of technologies, processes, and practices designed to protect networks, devices, programs, and data from attack, theft, damage, modification or unauthorized access. It may also be referred to as **information technology security**.

We live in a digital era which understands that our private information is more vulnerable than ever before. We all live in a world which is networked together, from internet banking to government infrastructure, where data is stored on computers and other devices. A portion of that data can be sensitive information, whether that be intellectual property, financial data, personal information, or other types of data for which unauthorized access or exposure could have negative consequences.

Cyber-attack is now an international concern and has given many concerns that hacks and other security attacks could endanger the global economy. Organizations transmit sensitive data across networks and to other devices in the course of doing businesses, and cybersecurity describes to protect that information and the systems used to process or store it.

As the volume of cyber-attacks grows, companies and organizations, especially those that deal information related to national security, health, or financial records, need to take steps to protect their sensitive business and personal information.

1.2 SCOPE OF THE PROJECT

The main objective of the project is to provide a location privacy of the user in the CBIR image based system. With the pervasive connectivity and availability of the Internet and digital imaging technologies, massive digital images are created and used in various fields, including medicine, publishing, education, and so on. Facing big image datasets, effective retrieval becomes a fundamental requirement. Currently, two image retrieval approaches dominate text based and content-based. In text-based image retrieval, images are searched based on the text descriptions associated with the images. In content-base image retrieval (CBIR), images are retrieved according to their visual similarities measured on low-level visual features. CBIR still works when textual annotations are not available, and has been implemented in the state-of-art image search engines, such as Google Image Search , and Bing Image Search . Considering the rich sensitive information embedded in images, recently, image privacy has been drawing more and more attention. Service provider owns an enormously huge number of images and is willing to provide the CBIR service on the dataset. Service provider can provide CBIR service via its own servers or outsourcing to cloud provider. Facing an untrustworthy service provider, the problem is how to prevent it from inferring the user's search intention. To our best knowledge, literatures are the only two research efforts in this category. Thus the image location gets changed according to the user privacy.

1.3 LITERATURE SURVEY

1.TITLE: ENCRYPTION-FREE FRAMEWORK OF PRIVACY-PRESERVING IMAGE RECOGNITION FOR PHOTO-BASED INFORMATION SERVICES

AUTHOR: Kazuaki Nakamura, Member, IEEE, Naoko Nitta, Member, IEEE, and Noboru Babaguchi

DESCRIPTION:

Nowadays mobile devices such as smartphones have been widely used all over the world. In addition, the performance of image recognition has drastically increased with deep learning technologies. From these backgrounds, some photo-based information services provided in a client-server architecture are getting popular: client users take a photo of a certain spot and send it to a server, while the server identifies the spot with an image recognizer and returns its related information to the users. However, this kind of client-server image recognition can cause a privacy issue because image recognition results are sometimes privacy-sensitive. To tackle the privacy issue, in this paper, we propose a framework of privacy-preserving image recognition called EnfPire, in which the server cannot uniquely determine the recognition result but client users can do so. An overview of EnfPire is as follows: First client users extract a visual feature from their taken photo and transform it so that the server cannot uniquely determine the recognition result. Then the users send the transformed feature to the server, which returns a set of candidates of the recognition result to the users. Finally, the users compare the candidates to the original visual feature for obtaining the final result. Our experimental results demonstrate that EnfPire successfully degrades the server's spot-recognition accuracy from 99.8% to 41.4% while keeping 86.9% of the spotrecognition accuracy on the user side

2. TITLE: SMARTSEC: SECRET SHARING-BASED LOCATION-AWARE PRIVACY ENHANCEMENT IN SMART DEVICES

AUTHOR: Bett Ben Chirchir, Xiaokuan Zhang, Mengyuan Li, Qiyang Qian, Na Ruan, Haojin Zhu

DESCRIPTION:

In the recent past we have witnessed a quick rise of smart devices, including smart phones, tablets, smart watch, smart television as well as other smart devices. The smart devices store quite a lot of sensitive personal data (e.g., private photos). These devices can breach one's privacy if information in the devices are accessed by unauthorized users. Therefore, smart devices are facing threats of privacy leaking and how to protect privacy of smart devices is regarded as an important research challenge. In this paper,

we propose a system, coined as SmartSec, which employs secret sharing techniques among multiple smart devices to enhance privacy protection. In particular, SmartSec can achieve privacy enhancement by storing sensitive data (e.g., photos) on at least three devices and requiring at least two of them to access the original data. This will thus secure the data from any unauthorized attackers because any leakage of one piece of secret is meaningless. Further, secret sharing will introduce extra computation and transmission overhead. To reduce such an overhead and optimize the performance, another important feature of SmartSec is that it could recognize and classify different locations by exploiting location tags and then apply different security policy to different locations to achieve the tradeoff of security and performance. For example, a user can demarcate safe regions (e.g., home area) and the sensitive data will be automatically recovered in home region, distributed and stored/masked at different devices in public regions, where the process is automated which provides the user a transparent experience. We implement the system and conducted a series of experiments on the prototype to evaluate the results.

3. TITLE: RECENT ADVANCES IN FEATURES EXTRACTION AND DESCRIPTION ALGORITHMS: A COMPREHENSIVE SURVEY

AUTHOR: Ehab Salahat, and Murad Qasaimeh

DESCRIPTION:

Computer vision is one of the most active research fields in information technology today. Giving machines and robots the ability to see and comprehend the surrounding world at the speed of sight creates endless potential applications and opportunities. Feature detection and description algorithms can be indeed considered as the retina of the eyes of such machines and robots. However, these algorithms are typically computationally intensive, which prevents them from achieving the speed of sight real-time performance. In addition, they differ in their capabilities and some may favor and work better given a specific type of input compared to others. As such, it is essential to compactly report their pros and cons as well as their performances and recent advances. This paper is dedicated to provide a comprehensive overview on the state-of-the-art and recent advances in feature detection and description algorithms. Specifically, it starts by over-viewing fundamental concepts. It then compares, reports and discusses their

performance and capabilities. The Maximally Stable Extremal Regions algorithm and the Scale Invariant Feature Transform algorithms, being two of the best of their type, are selected to report their recent algorithmic derivatives.

4. TITLE: AN ANALOGY-RELEVANCE FEEDBACK CBIR METHOD USING MULTIPLE FEATURES

AUTHOR: Hui Xie, Ying Ji, Yueming Lu

DESCRIPTION:

Since traditional relevance feedback content based image retrieval (CBIR) methods need several rounds of search, in this paper we put forward an analogy-relevance feedback (analogy-RF) CBIR method using multiple features which only needs one. The method allows users to choose the kind of object of the query image when they input the query image, and our system can determine several analogy-RF images in the sample database. Then we can use analogy-RF images to revise the similarity of images and retrieval and sort images by the re-calculated similarity. The experiment result on COREL 1k image database shows the effectiveness of the proposed method

5. TITLE: THE GEO-PRIVACY BONUS OF POPULAR PHOTO ENHANCEMENTS

AUTHOR: Jaeyoung Choi, Martha Larson and Xinchao Li

DESCRIPTION:

Today's geo-location estimation approaches are able to infer the location of a target image using its visual content alone. These approaches typically exploit visual matching techniques, applied to a large collection of background images with known geo-locations. Users who are unaware that visual analysis and retrieval approaches can compromise their geo-privacy, unwittingly open themselves to risks of crime or other unintended consequences. This paper lays the groundwork for a new approach to geo-

privacy of social images: Instead of requiring a change of user behavior, we start by investigating users' existing photo-sharing practices. We carry out a series of experiments using a large collection of social images to systematically analyze how photo editing practices impact the performance of geo-location estimation. We find that standard image enhancements, including filters and cropping, already serve as natural geo-privacy protectors. In our experiments, up to 19% of images whose location would otherwise be automatically predictable were unlocalizeable after enhancement. We conclude that it would be wrong to assume that geo-visual privacy is a lost cause in today's world of rapidly maturing machine learning. Instead, protecting users against the unwanted effects of pixel-based inference is a viable research field. A starting point understands the geo-privacy bonus of already established user behavior.

CHAPTER 2

AIM AND SCOPE OF THE PRESENT INVESTIGATION

2.1 AIM OF THE PROPOSED SYSTEM

The user privacy of the location from the images is achieved by the AES based encryption scheme where the location as a cipher text in the server. The main challenge is that the location can be hacked by the hackers after that of the encryption get achieved. Thus the key based encryption is implemented. Firstly, the image owner calculates the order or disorder difference matrices of RGB components and encrypts them by value replacement and position scrambling. The encrypted images are then uploaded to the cloud server who extracts encrypted difference histograms as image feature vectors. To search similar images, the query image is encrypted by the image users as the image owner does, and the query feature vector is extracted by the cloud server. The Euclidean distance between query feature vector and image feature vector is calculated to measure the similarity. The security analysis and experiments demonstrate the usability of the proposed scheme

The main problem of the existing system is that the encryption process. The aim of the existing project is to protect the sensitive information from the image that leaks the user privacy. Thus the encryption is implemented to save the details from the third party. But the data can be leaked into the server since the encryption is made only after the data get stored in the server. If you directly share an image file with someone, you are also including the embedded metadata. This means if you email it to them, text it to them, or use a shared folder in a service like Dropbox or Google Drive, they will have the GPS coordinates where you took it. This might sound scary, but let's back things up. Thus the location privacy plays a major issue in the image sharing, uploading on social media and so on. Thus the image upload can be made by the encryption free framework such that the data can be automatically saved in a secured manner without encryption process.

2.2 SCOPE OF THE PROPOSED SYSTEM

In this project it is proposed that an encryption free mechanism for the image recognition of the location, which we call EnfPire, inspired by a privacy preserving framework for image retrieval. EnfPire does not impose any restrictions on the server's recognition algorithm because it does not rely on cryptographic techniques. In EnfPire, the server cannot identify the client users' current location; its candidates can only be presented. This can successfully abstract the location information, but is not enough for protecting its history. When the users continually use this kind of photo-based information service several times, their location history could be guessed by reducing the candidates of the user locations based on the geographical relationship between spots, which can be typically obtained from a map of the field. We refer to this kind of attack as map-based attack (MBA). To make EnfPire robust to MBA, we also propose its countermeasure. Private Content Based Image Retrieval (PCBIR) deals with the retrieval of similar content without revealing the content of the query image to the database. PCBIR is achieved by exchange of messages between the user and the database. These messages collectively help the user in inferring the required information from the database but prohibit the database from knowing the user's interest. In this sense, PCBIR is similar to private information retrieval (PIR) schemes that allow a user to obtain the data stored at a specific address in a database whilst keeping the database oblivious of the address.

ADVANTAGES OF THE PROPOSED SYSTEM

- The advantages of the system will be made use of the location privacy enhanced
- The privacy can be achieved using the EnfPire system
- The privacy achieved at good level so that the hacker will be in a sophisticated mood.

2.3 EXISTING SYSTEM

A CBIR based system is implemented where the content of the image gets retrieved when they are get uploaded to the server. Visual contents such as images and video generally have two types of privacy-sensitive information: visual data itself and processing results of the visual data. Examples of the former include human faces, entire bodies, car license plates, and so on. The latter examples are results of content-based image retrieval (CBIR) and those of image recognition. Content-Based Image Retrieval (CBIR) systems have been used for the searching of relevant images in various research areas. In CBIR systems features such as shape, texture and color are used. The extraction of features is the main step on which the retrieval results depend. Color features in CBIR are used as in the color histogram, color moments, conventional color correlogram and color histogram. Color space selection is used to represent the information of color of the pixels of the query image. The shape is the basic characteristic of segmented regions of an image. Different methods are introduced for better retrieval using different shape representation techniques; earlier the global shape representations were used but with time moved towards local shape representations. The local shape is more related to the expressing of result instead of the method. Local shape features may be derived from the texture properties and the color derivatives. Texture features have been used for images of documents, segmentation-based recognition and satellite images. Texture features are used in different CBIR systems along with color, shape, geometrical structure and sift features. Thus to produce a location privacy for the content a encryption based technology is implemented. An AES based encryption algorithm protects the private information from the images.

DISADVANTAGES

- The location privacy can be implemented only after the location get stored in the server
- Thus when the location is stored once the hacker can easily hack the location of the user.

CHAPTER 3

SYSTEM CONFIGURATION

3.1 SYSTEM ARCHITECTURE

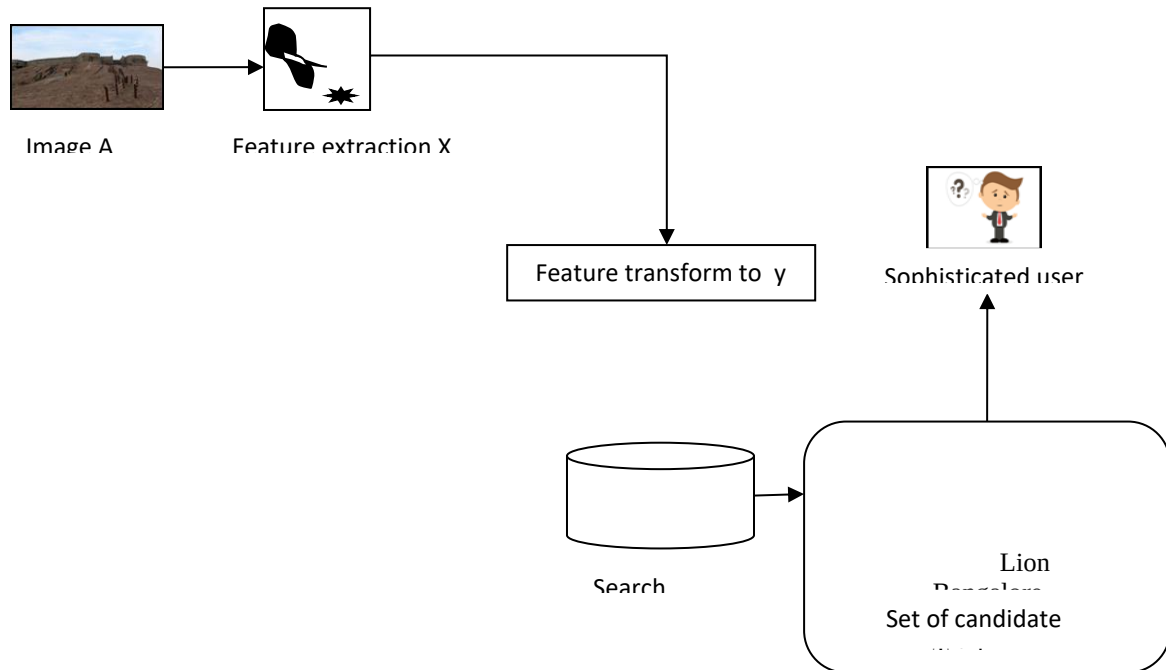


Figure 1 : System Architecture

3.2 HARDWARE REQUIREMENTS

- Processor : Dual core processor 2.6.0 GHz
- RAM : 1GB
- Hard disk : 160 GB
- Compact Disk : 650 MB
- Keyboard : Standard keyboard
- Monitor : 15 inch color monitor

3.3 SOFTWARE REQUIREMENTS

- Front End : JAVA

- Back End : SQL
- Platform : Windows 7

CHAPTER 4

EXPERIMENTS, METHODS AND ALGORITHM

4.1 METHODOLOGY

- User registration and login
- Feature extraction
- Transform features

4.1.1 USER REGISTRATION AND LOGIN

In user registration module the user will register their entire details where the user can do the process. The user has to provide his entire details such that the user details will be saved in the database. From that the username and the password is provide for that particular user authentication system.

4.1.2 DATA FLOW DIAGRAM OF USER LOGIN CREDENTIALS

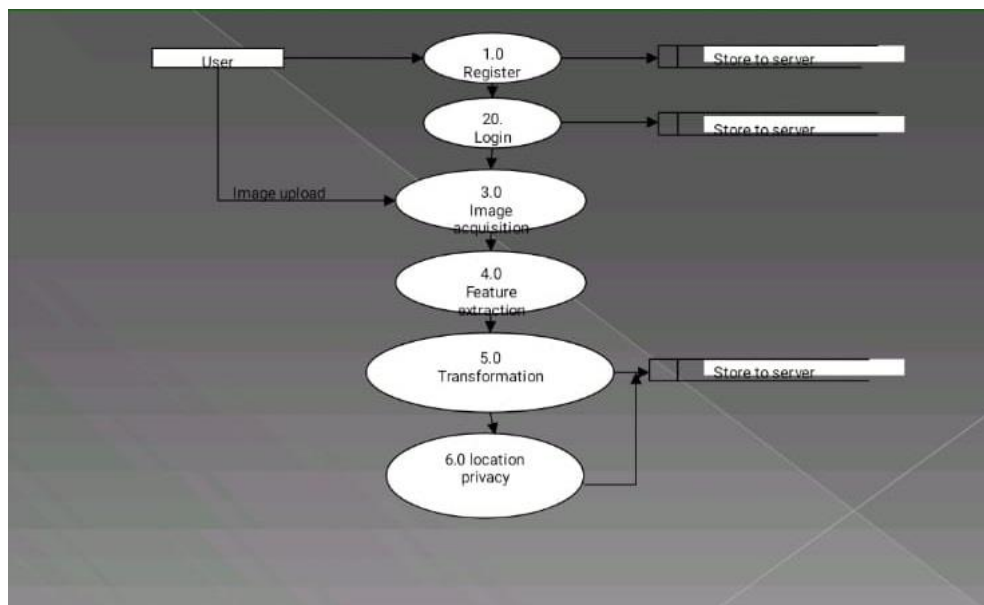


Figure : user login credentials

4.1.3 FEATURE EXTRACTION

Feature extraction is a process of dimensionality reduction by which an initial set of raw data is reduced to more manageable groups for processing. A characteristic of these large data sets is a large number of variables that require a lot of computing resources to process. In general, feature extraction is an essential processing step in pattern recognition and machine learning tasks. The goal is to extract a set of features from the dataset of interest. These features must be informative with respect to the desired properties of the original data. Feature extraction is a dimensionality reduction process, where an initial set of raw variables is reduced to more manageable groups (features) for processing, while still accurately and completely describing the original data set. In machine learning, pattern recognition and in image processing, feature extraction starts from an initial set of measured data and builds derived values (features) intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps

The feature extraction can be made by the SIFT and the CNN techniques thus this made an efficient implementation of the extraction of the feature from the images. Such, that the efficient method is enhanced for the precision accuracy of the system.

4.1.4 TRANSFORM FEATURES

After the feature get extracted all the extracted features are kept aside. A pooling based implementation is added where the needed feature is nothing but the location from the images. The location which is extracted needed to be modified based on the transform features. The transformed features and their efficiency are placed at a negative impact. So that the extracted features is added with a features extracted from the other location or spot founded. The spot can be modified in the server where the server cannot able to view the data. Thus the transformed features are made in this system to provide location privacy for the user private information's.

4.2 ALGORITHM USED

Two main algorithms implemented here is Scale Invariant Feature Transformation (SIFT) and the Convolution Neural Network (CNN) for the feature extraction.

4.2.1 SIFT TECHNIQUE

The Scale Invariant Feature Transform (SIFT) is used for interest region detector using Difference of Gaussian (DoG) and feature descriptor. The SIFT feature descriptor achieves the robustness to various illumination, lighting and positional shifts by encoding in a localized set of gradient orientation histograms (HoG) . In the first step, the gradient magnitude and orientation of image is examined around the key point location to select the level of Gaussian Blur using the region scale. In each examined region, sampling is performed in the 16×16 area of regular grid, which is covering the interest region. The gradient orientation is entered into the 4×4 patch of gradient histogram with 8 bins each. The main reason of this Gaussian window is to give higher weight to pixels closer to middle regions which are less position variant. Once all the histogram entries have been completed then those entries are concatenated to form a single feature vector with 128 dimensions, i-e $4 \times 4 \times 8 = 128$. Finally, all the values to normalized to the unit vector to minimize the influence of high spikes in the histogram. A SIFT keypoint is a circular image region with an orientation. ... SIFT keypoints are circular image regions with an orientation. There are several parameters that influence the detection of SIFT keypoints. First, searching keypoints at multiple scales is obtained by constructing a so-called "Gaussian scale space".

4.2.1.1 GAUSSIAN PYRAMID

The first stage is to construct a Gaussian "scale space" function from the input image [1]. This is formed by convolution (filtering) of the original image with Gaussian functions of varying widths. The difference of Gaussian (DoG), $D(x, y, \sigma)$, is calculated as the difference between two filtered images, one with k multiplied by scale of the other.

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (1)$$

These images, $L(x, y, \sigma)$, are produced from the convolution of Gaussian functions, $G(x, y, k\sigma)$, with an input image, $I(x, y)$.

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (2)$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\{-(x^2 + y^2)/2\sigma^2\} \quad (3)$$

This is the approach we use in the implementation. First, the initial image, I , is convolved with a Gaussian function, G_0 , of width σ_0 . Then we use this blurred image, L_0 , as the first image in the Gaussian pyramid and incrementally convolve it with a Gaussian, G_i , of width σ_i to create the i th image in the image pyramid, which is equivalent to the original image filtered with a Gaussian, G_k , of width $k\sigma_0$. The effect of convolving with two Gaussian functions of different widths is most easily found by converting to the Fourier domain, in which convolution becomes multiplication. Once a complete octave has been processed, we subsample the Gaussian image that has twice the initial value of σ by taking every second pixel in each row and column. This greatly improves the efficiency of the algorithm at lower scales.

4.2.1.2 EXTREMA DETECTION

This stage is to find the extrema points in the DOG pyramid. To detect the local maxima and minima of $D(x, y, \sigma)$, each point is compared with the pixels of all its 26 neighbours. If this value is the minimum or maximum this point is an extrema. We then improve the localisation of the keypoint to subpixel accuracy, by using a second order Taylor series expansion. This gives the true extrema location as:

$$z = - \left(\frac{\partial^2 D}{\partial x^2} \right) \frac{\partial D}{\partial x} \quad (4)$$

where D and its derivatives are evaluated at the sample point and $x = x_0 + \Delta x$, $y = y_0 + \Delta y$, $\sigma = \sigma_0 + \Delta \sigma$ is the offset from the sample point.

4.2.1.3 CONVOLUTION NEURAL NETWORK (CNN)

Convolutional neural network (CNN) is a machine learning method which has successfully been applied to the field of image classification. The structure roughly mimics the nature of the mammalian visual cortex and neural networks in the brain. It is inspired by the human visual system because of its ability to recognize and localize objects within cluttered scenes. That ability is desired within artificial system in order to overcome the challenges of recognizing objects in a class despite high in-class variability and perspective variability.

Convolutional neural networks is a form of artificial neural networks. The structure of an artificial neural network is shown in figure 5.2.1.

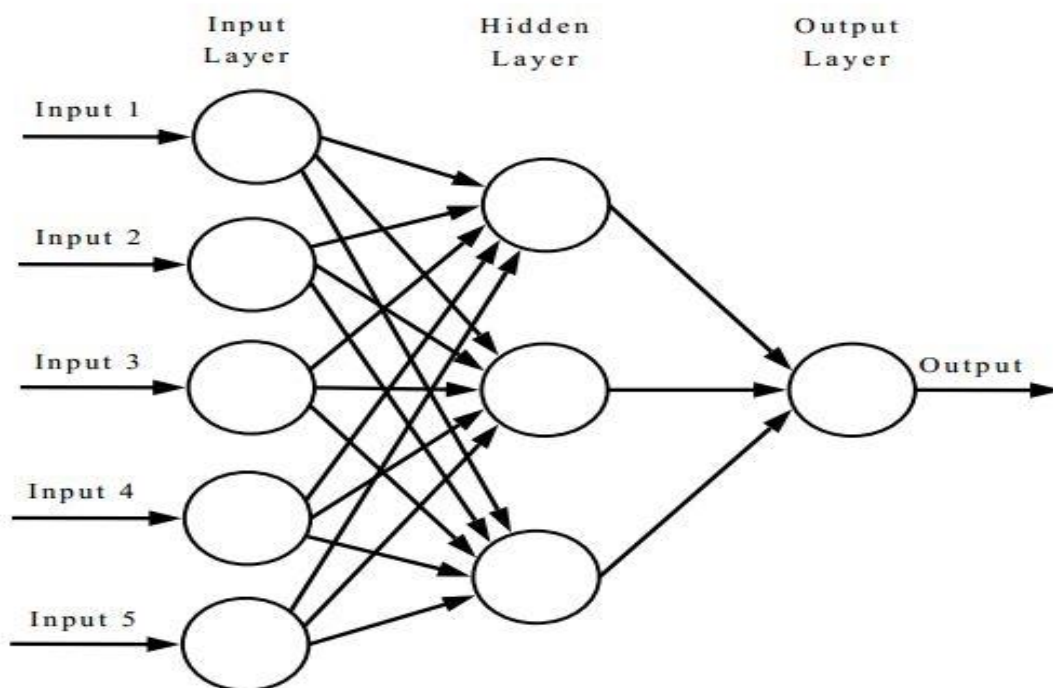


Fig 3.1The structure of an artificial neural network.

An artificial neural network consists of neurons in multiple layers; the input layer, the output layer and one or more hidden layers. Networks with two or more hidden layers are called deep neural networks. The input layer consists of an input data and the output

layer consists of a value indicating whether the neuron is activated or not. In the case of classification, the neurons in the output layer represent the different classes. Each of the neurons in the output layer results in a soft-max value which describes the probability of the input belonging to that class. The input to a neuron is the weighted outputs of the neurons in the previous layer, if a layer is fully connected it consists of the output from all neurons in the previous layer. The weight controls the amount of influence the output of a neuron has on the next neuron. The hidden layers each consists of different combinations of the weighted outputs of the previous layers. That way, with increased number of hidden layers, more complex decisions can be made. The method can simplified be described as composing complex combinations of the information about the input data which correctly maps the input data to the correct output. In the training part, when the network is trained, those complex combinations are formed, which can be thought of as a classification model. In the evaluation part, that model is used to classify new data.

CHAPTER 5

RESULT AND DISCUSSION

5.1 DISCUSSION

Each of the classifications and the retrieval is evaluated separately. For binary classification the resulting output for every image is either the positive or the negative class which is either true or false. This means each image can be described as a true/false positive/negative. For the retrieval part, the resulting output for each image is whether it should be retrieved or not, which is either true or false. This means that every image can be described as a true/false negative/positive. After evaluating each part separately the system is put together. For each of the classifications, the feature extraction method which provided the best resulting average accuracy is used

5.2 RESULT

. The results of the entire system are then evaluated. That is done by describing which images are retrieved as worthy of further analysis and how well it conforms to which images that should be. Images that are worthy of further analysis are images that are good, salient and unique with respect to the other retrieved images. The final output for an image is whether its retrieval is true or false, the same way as for the retrieval part. That way, true/false negatives/positives are achieved.

All results will be evaluated using the measures precision, recall and accuracy which are defined as:

$$\text{Accuracy} = \frac{\text{true positives} + \text{true negatives}}{\text{all samples}} \quad (8)$$

Precision and average precision: To more clearly record the retrieval performance, average precision (AP) is used, which amounts to area under the precision recall curve. Typically, a larger AP means a higher precision recall curve and thus better retrieval performance. Since retrieval datasets typically have multiple queries, their respective APs are averaged to produce a final performance evaluation.

5.2.1 UNIT TESTING

The first test in the development process is the unit test. The source code is normally divided into modules, which in turn are divided into smaller units called units. These units have specific behavior. The test done on these units of code is called unit test. Unit test depends upon the language on which the project is developed. Unit tests ensure that each unique path of the project performs accurately to the documented specifications and contains clearly defined inputs and expected results. Functional and reliability test in an Engineering environment producing tests for the behavior of components (nodes and vertices) of a product to ensure their correct behavior prior to system integration.

5.2.2 SYSTEM TESTING

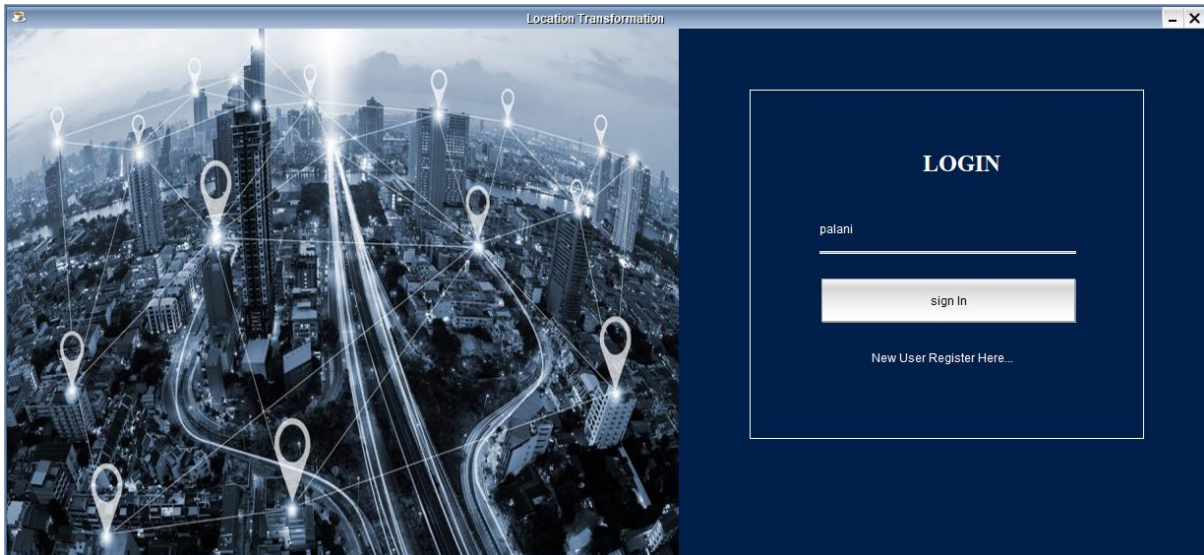
Several modules constitute a project. If the project is long-term project, several developers write the modules. Once all the modules are integrated, several errors may arise. The testing done at this stage is called system test. System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points. Testing a specific hardware/software installation. This is typically performed on a COTS (commercial off the shelf) system or any other system comprised of disparate parts where custom configurations and/or unique installations are the norm.

5.2.3 INTEGRATION TESTING

Testing in which modules are combined and tested as a group. Modules are typically code modules, individual applications, source and destination applications on a network, etc. Integration Testing follows unit testing and precedes system testing. Testing after the product is code complete. Betas are often widely distributed or even distributed to the public at large in hopes that they will buy the final product when it is release.

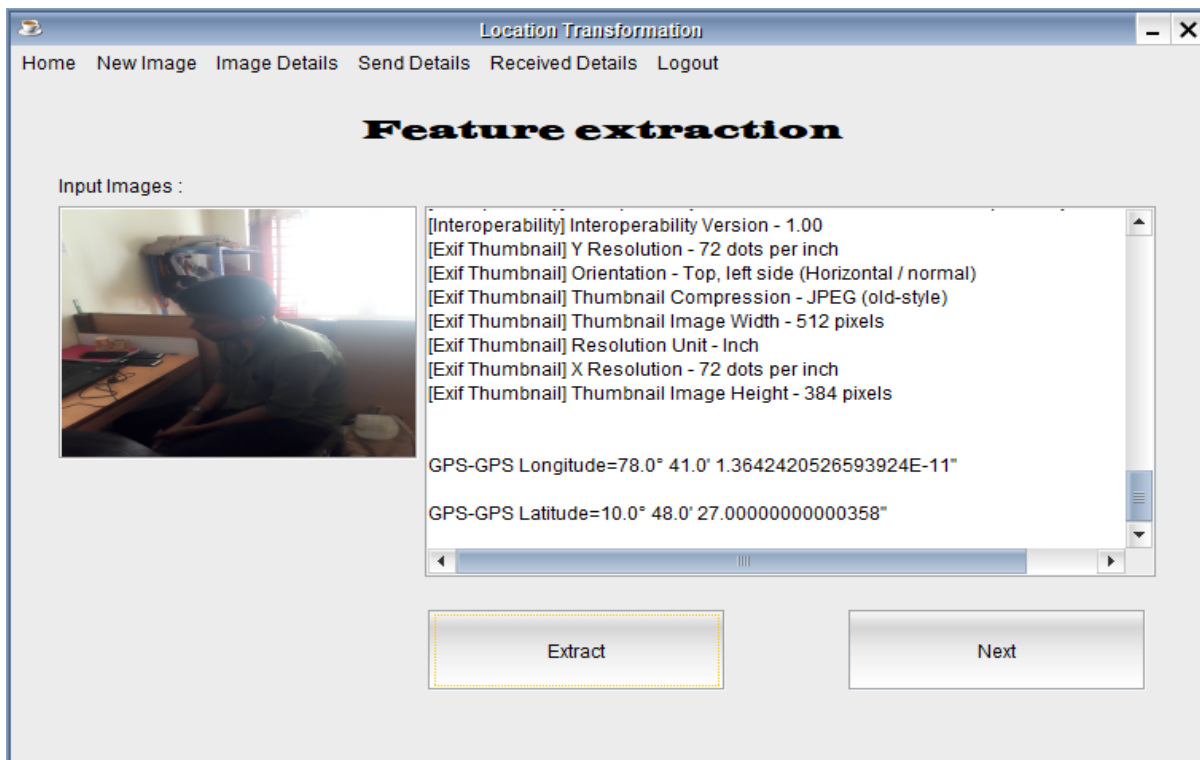
5.3 TEST SCREENS

UNIT TESTING



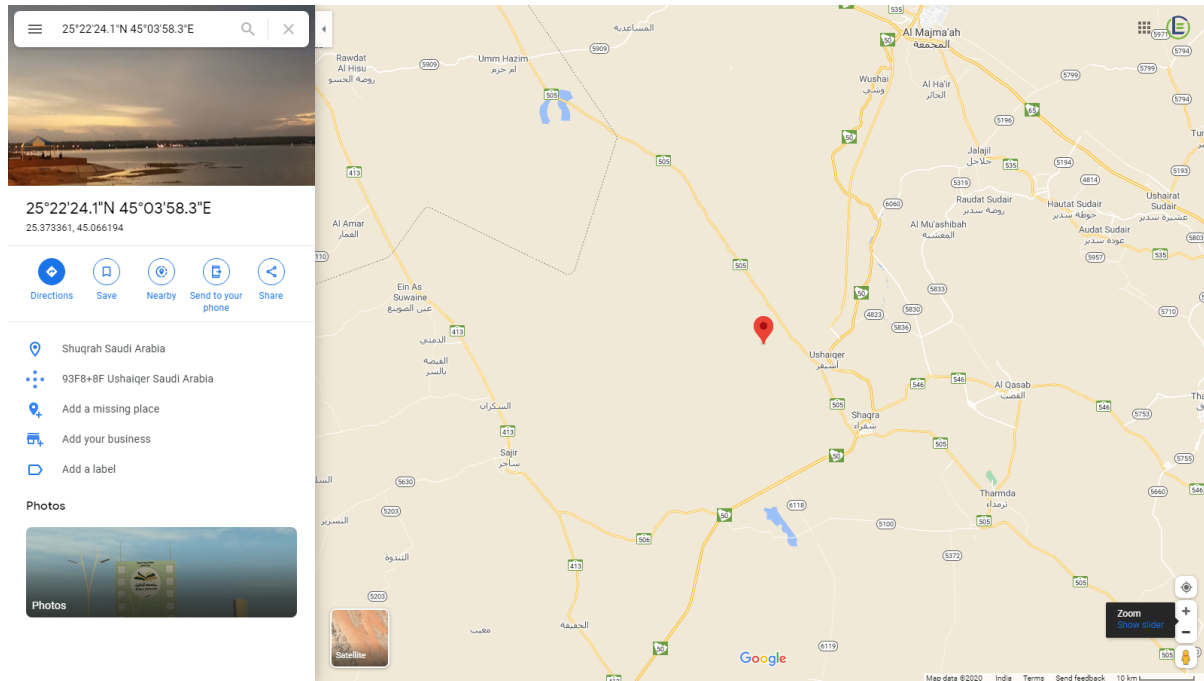
User Login

INTEGRATION TESTING



Feature extraction verification

SYSTEM TESTING



Location extraction

VALIDATION TESTING

The screenshot shows a web application window with the title "Location Transformation". Inside the window, there is a dark blue background with a white border. At the top center, the word "REGISTRATION" is written in white, bold, uppercase letters. Below this, there are two columns of input fields. The left column contains three fields: "Name" with the value "palani", "Email" with the value "palani@gmail.com", and "Username" with the value "palani". The right column contains three fields: "Contact" with the value "9876543210", "Address" with the value "trichy", and "Password" with the value "3414". Each field has a white underline. At the bottom center, there is a white button with the text "Register" in black.

Name	Contact
palani	9876543210
Email	Address
palani@gmail.com	trichy
Username	Password
palani	3414

Register

User registration

CHAPTER 6

CONCLUSION

In this project we have addressed the problem of private retrieval in image databases. The algorithm is shown to be customizable for hierarchical data structures as well as hash-based indexing schemes. Experimental study has proved that the algorithm is accurate, efficient and scalable. We proposed algorithms that are fully private and feasible on reasonably large databases using a variety of state of the art indexing schemes. We have also demonstrated that image databases are amenable to significantly faster private retrieval by exploiting the property of image data. Specifically we demonstrated a near-linear operating region for image databases, where the trade off between privacy and speed is feasible. We provided a theoretical analysis on privacy protection of the new scheme. It shows the new scheme can effectively control privacy leakage and significantly reduce the attack success probability. Moreover, we carried out a large number of experiments on a realworld image collection. The results demonstrate that the privacy preserving performance of PBCBIR is significantly improved compared to RF-CBIR, while the retrieval performance sacrifice is acceptable. The success rate of feedback attack with CNN drops dramatically from 100% in RF-CBIR to 41% in PB-CBIR ($K=5$), while the average precision of top 20 decreases slightly from 46% to 42%. We also investigated the parameters of the new scheme and give the suggestions. The location of the CBIR image gets protected in the process

6.1 APPENDIX

6.2 SOURCE CODING

Home page

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<Form version="1.3" maxVersion="1.9"  
type="org.netbeans.modules.form.forminfo.JFrameFormInfo">
```

```
<Properties>
```

```
<Property name="defaultCloseOperation" type="int" value="3"/>
```

```
<Property name="preferredSize" type="java.awt.Dimension"  
editor="org.netbeans.beaninfo.editors.DimensionEditor">
```

```
<Dimension value="[1224, 564]"/>
```

```
</Property>
```

```
</Properties>
```

```
<SyntheticProperties>
```

```
<SyntheticProperty name="formSizePolicy" type="int" value="1"/>
```

```
<SyntheticProperty name="generateCenter" type="boolean" value="false"/>
```

```
</SyntheticProperties>
```

```
<AuxValues>
```

```
<AuxValue name="FormSettings_autoResourcing" type="java.lang.Integer" value="0"/>
```

```
<AuxValue name="FormSettings_autoSetComponentName" type="java.lang.Boolean"  
value="false"/>
```

```
<AuxValue name="FormSettings_generateFQN" type="java.lang.Boolean"  
value="true"/>
```

```
<AuxValue name="FormSettings_generateMnemonicsCode" type="java.lang.Boolean"  
value="false"/>
```

```

<AuxValue name="FormSettings_i18nAutoMode" type="java.lang.Boolean"
value="false"/>

<AuxValue name="FormSettings_layoutCodeTarget" type="java.lang.Integer"
value="1"/>

<AuxValue name="FormSettings_listenerGenerationStyle" type="java.lang.Integer"
value="0"/>

<AuxValue name="FormSettings_variablesLocal" type="java.lang.Boolean"
value="false"/>

<AuxValue name="FormSettings_variablesModifier" type="java.lang.Integer"
value="2"/>

</AuxValues>

```

```

<Layout>

<DimensionLayout dim="0">

<Group type="103" groupAlignment="0" attributes="0">

<Component id="jPanel1" min="-2" max="-2" attributes="0"/>

</Group>

</DimensionLayout>

<DimensionLayout dim="1">

<Group type="103" groupAlignment="0" attributes="0">

<Component id="jPanel1" max="32767" attributes="0"/>

</Group>

</DimensionLayout>

</Layout>

<SubComponents>

<Container class="javax.swing.JPanel" name="jPanel1">

<Properties>

```

```
<Property name="background" type="java.awt.Color"
editor="org.netbeans.beaninfo.editors.ColorEditor">
```

```
<Color blue="4b" green="1f" red="0" type="rgb"/>
```

```
</Property>
```

```
</Properties>
```

```
<Layout>
```

```
<DimensionLayout dim="0">
```

```
<Group type="103" groupAlignment="0" attributes="0">
```

```
<Group type="102" alignment="1" attributes="0">
```

```
<Component id="l1" pref="683" max="32767" attributes="0"/>
```

```
<EmptySpace max="-2" attributes="0"/>
```

```
<Component id="jPanel2" min="-2" max="-2" attributes="0"/>
```

```
<EmptySpace max="-2" attributes="0"/>
```

```
</Group>
```

```
</Group>
```

```
</DimensionLayout>
```

```
<DimensionLayout dim="1">
```

```
<Group type="103" groupAlignment="0" attributes="0">
```

```
<Component id="jPanel2" alignment="0" max="32767" attributes="0"/>
```

```
<Group type="102" alignment="0" attributes="0">
```

```
<Component id="l1" min="-2" pref="574" max="-2" attributes="0"/>
```

```
<EmptySpace min="0" pref="0" max="32767" attributes="0"/>
```

```
</Group>
```

```
</Group>
```

```
</DimensionLayout>
```

```

</Layout>

<SubComponents>

<Container class="javax.swing.JPanel" name="jPanel2">

<Properties>

<Property name="background" type="java.awt.Color"
editor="org.netbeans.beaninfo.editors.ColorEditor">

<Color blue="4b" green="1f" red="0" type="rgb"/>

</Property>

</Properties>


<Layout>

<DimensionLayout dim="0">

<Group type="103" groupAlignment="0" attributes="0">

<Group type="102" alignment="0" attributes="0">

<EmptySpace min="-2" pref="60" max="-2" attributes="0"/>

<Component id="jPanel3" min="-2" max="-2" attributes="0"/>

<EmptySpace pref="66" max="32767" attributes="0"/>

</Group>

</Group>

</DimensionLayout>

<DimensionLayout dim="1">

<Group type="103" groupAlignment="0" attributes="0">

<Group type="102" alignment="0" attributes="0">

<EmptySpace min="-2" pref="62" max="-2" attributes="0"/>

<Component id="jPanel3" min="-2" max="-2" attributes="0"/>

<EmptySpace max="32767" attributes="0"/>

```

```

</Group>

</Group>

</DimensionLayout>

</Layout>

<SubComponents>

<Container class="javax.swing.JPanel" name="jPanel3">

<Properties>

<Property name="background" type="java.awt.Color"
editor="org.netbeans.beaninfo.editors.ColorEditor">

<Color blue="4b" green="1f" red="0" type="rgb"/>

</Property>

<Property name="border" type="javax.swing.border.Border"
editor="org.netbeans.modules.form.editors2.BorderEditor">

<Border info="org.netbeans.modules.form.compat2.border.LineBorderInfo">

<LineBorder>

<Color PropertyName="color" blue="ff" green="ff" red="ff" type="rgb"/>

</LineBorder>

</Border>

</Property>

</Properties>

<Layout>

<DimensionLayout dim="0">

<Group type="103" groupAlignment="0" attributes="0">

<Group type="102" alignment="0" attributes="0">

<Group type="103" groupAlignment="1" attributes="0">

<Group type="102" alignment="1" attributes="0">

```

```

<EmptySpace max="-2" attributes="0"/>
<Component id="name" min="-2" pref="261" max="-2" attributes="0"/>
</Group>
<Group type="102" alignment="0" attributes="0">
<EmptySpace min="-2" pref="70" max="-2" attributes="0"/>
<Group type="103" groupAlignment="1" attributes="0">
<Group type="103" groupAlignment="0" attributes="0">
<Component id="jButton2" alignment="0" min="-2" pref="259" max="-2" attributes="0"/>
<Group type="102" alignment="1" attributes="0">
<EmptySpace pref="51" max="-2" attributes="0"/>
<Component id="jLabel3" min="-2" pref="168" max="-2" attributes="0"/>
<EmptySpace min="-2" pref="40" max="-2" attributes="0"/>
</Group>
</Group>
<Component id="jSeparator1" min="-2" pref="261" max="-2" attributes="0"/>
</Group>
</Group>
</Group>
<EmptySpace pref="68" max="32767" attributes="0"/>
</Group>
<Group type="103" rootIndex="1" groupAlignment="0" attributes="0">
<Group type="102" alignment="0" attributes="0">
<EmptySpace min="-2" pref="175" max="-2" attributes="0"/>
<Component id="jLabel1" min="-2" max="-2" attributes="0"/>
<EmptySpace pref="145" max="32767" attributes="0"/>
</Group>

```

```

</Group>
</Group>
</DimensionLayout>
<DimensionLayout dim="1">
<Group type="103" groupAlignment="0" attributes="0">
<Group type="102" alignment="0" attributes="0">
<EmptySpace min="-2" pref="124" max="-2" attributes="0"/>
<Component id="name" min="-2" pref="33" max="-2" attributes="0"/>
<EmptySpace max="-2" attributes="0"/>
<Component id="jSeparator1" min="-2" pref="10" max="-2" attributes="0"/>
<EmptySpace type="separate" max="-2" attributes="0"/>
<Component id="jButton2" min="-2" pref="45" max="-2" attributes="0"/>
<EmptySpace type="separate" max="-2" attributes="0"/>
<Component id="jLabel3" min="-2" pref="34" max="-2" attributes="0"/>
<EmptySpace pref="65" max="32767" attributes="0"/>
</Group>
<Group type="103" rootIndex="1" groupAlignment="0" attributes="0">
<Group type="102" alignment="0" attributes="0">
<EmptySpace min="-2" pref="58" max="-2" attributes="0"/>
<Component id="jLabel1" min="-2" pref="32" max="-2" attributes="0"/>
<EmptySpace pref="263" max="32767" attributes="0"/>
</Group>
</Group>
</Group>
</DimensionLayout>
</Layout>

```

```

<SubComponents>

<Component class="javax.swing.JSeparator" name="jSeparator1">

</Component>

<Component class="javax.swing.JLabel" name="jLabel3">

<Properties>

<Property name="foreground" type="java.awt.Color"
editor="org.netbeans.beaninfo.editors.ColorEditor">

<Color blue="ff" green="ff" red="ff" type="rgb"/>

</Property>

<Property name="text" type="java.lang.String" value="New User Register Here..."/>

<Property name="toolTipText" type="java.lang.String" value="Create a New Account"/>

</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jLabel3MouseClicked"/>

</Events>

</Component>

<Component class="javax.swing.JLabel" name="jLabel1">

<Properties>

<Property name="font" type="java.awt.Font"
editor="org.netbeans.beaninfo.editors.FontEditor">

<Font name="Times New Roman" size="24" style="1"/>

</Property>

<Property name="foreground" type="java.awt.Color"
editor="org.netbeans.beaninfo.editors.ColorEditor">

<Color blue="ff" green="ff" red="ff" type="rgb"/>

</Property>

```

```

<Property name="text" type="java.lang.String" value="LOGIN"/>
</Properties>
</Component>
<Component class="javax.swing.JButton" name="jButton2">
<Properties>
<Property name="text" type="java.lang.String" value="sign In"/>
</Properties>
<Events>
<EventHandler event="actionPerformed" listener="java.awt.event.ActionListener"
parameters="java.awt.event.ActionEvent" handler="jButton2ActionPerformed"/>
</Events>
</Component>
<Component class="javax.swing.JTextField" name="name">
<Properties>
<Property name="background" type="java.awt.Color"
editor="org.netbeans.beaninfo.editors.ColorEditor">
<Color blue="4b" green="1f" red="0" type="rgb"/>
</Property>
<Property name="foreground" type="java.awt.Color"
editor="org.netbeans.beaninfo.editors.ColorEditor">
<Color blue="ff" green="ff" red="ff" type="rgb"/>
</Property>
<Property name="toolTipText" type="java.lang.String" value="Enter Your Name"/>
<Property name="border" type="javax.swing.border.Border"
editor="org.netbeans.modules.form.editors2.BorderEditor">
<Border info="null"/>
</Property>

```

```

</Properties>
</Component>
</SubComponents>
</Container>
</SubComponents>
</Container>
<Component class="javax.swing.JLabel" name="l1">
</Component>
</SubComponents>
</Container>
</SubComponents>
</Form>
<?xml version="1.0" encoding="UTF-8" ?>

<Form version="1.3" maxVersion="1.9"
type="org.netbeans.modules.form.forminfo.JFrameFormInfo">
<NonVisualComponents>
<Component class="javax.swing.JMenuItem" name="jMenuItem1">
<Properties>
<Property name="text" type="java.lang.String" value="jMenuItem1"/>
</Properties>
</Component>
<Menu class="javax.swing.JMenuBar" name="jMenuBar1">
<SubComponents>
<Menu class="javax.swing.JMenu" name="jMenu1">
<Properties>

```

```

<Property name="text" type="java.lang.String" value="Home"/>
</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu1MouseClicked"/>
</Events>

</Menu>

<Menu class="javax.swing.JMenu" name="jMenu2">

<Properties>

<Property name="text" type="java.lang.String" value="New Image"/>
</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu2MouseClicked"/>
</Events>

</Menu>

<Menu class="javax.swing.JMenu" name="jMenu3">

<Properties>

<Property name="text" type="java.lang.String" value="Image Details"/>
</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu3MouseClicked"/>
</Events>

</Menu>

<Menu class="javax.swing.JMenu" name="jMenu4">

<Properties>

```

```

<Property name="text" type="java.lang.String" value="Send Details"/>
</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu4MouseClicked"/>
</Events>

</Menu>

<Menu class="javax.swing.JMenu" name="jMenu5">

<Properties>

<Property name="text" type="java.lang.String" value="Received Details"/>
</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu5MouseClicked"/>
</Events>

</Menu>

<Menu class="javax.swing.JMenu" name="jMenu6">

<Properties>

<Property name="text" type="java.lang.String" value="Logout"/>
</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu6MouseClicked"/>
</Events>

</Menu>

</SubComponents>

</Menu>

```

```
</NonVisualComponents>

<Properties>

<Property name="defaultCloseOperation" type="int" value="3"/>

</Properties>

<SyntheticProperties>

<SyntheticProperty name="menuBar" type="java.lang.String" value="jMenuBar1"/>

<SyntheticProperty name="formSizePolicy" type="int" value="1"/>

<SyntheticProperty name="generateCenter" type="boolean" value="false"/>

</SyntheticProperties>

<AuxValues>

<AuxValue name="FormSettings_autoResourcing" type="java.lang.Integer" value="0"/>

<AuxValue name="FormSettings_autoSetComponentName" type="java.lang.Boolean"
value="false"/>

<AuxValue name="FormSettings_generateFQN" type="java.lang.Boolean"
value="true"/>

<AuxValue name="FormSettings_generateMnemonicsCode" type="java.lang.Boolean"
value="false"/>

<AuxValue name="FormSettings_i18nAutoMode" type="java.lang.Boolean"
value="false"/>

<AuxValue name="FormSettings_layoutCodeTarget" type="java.lang.Integer"
value="1"/>

<AuxValue name="FormSettings_listenerGenerationStyle" type="java.lang.Integer"
value="0"/>

<AuxValue name="FormSettings_variablesLocal" type="java.lang.Boolean"
value="false"/>

<AuxValue name="FormSettings_variablesModifier" type="java.lang.Integer"
value="2"/>

</AuxValues>
```

```

<Layout>
<DimensionLayout dim="0">
<Group type="103" groupAlignment="0" attributes="0">
<Group type="102" alignment="0" attributes="0">
<EmptySpace max="-2" attributes="0"/>
<Component id="jPanel1" max="32767" attributes="0"/>
<EmptySpace max="-2" attributes="0"/>
</Group>
</Group>
</DimensionLayout>
<DimensionLayout dim="1">
<Group type="103" groupAlignment="0" attributes="0">
<Group type="102" alignment="0" attributes="0">
<EmptySpace max="-2" attributes="0"/>
<Component id="jPanel1" max="32767" attributes="0"/>
<EmptySpace max="-2" attributes="0"/>
</Group>
</Group>
</DimensionLayout>
</Layout>
<SubComponents>
<Container class="javax.swing.JPanel" name="jPanel1">

<Layout>
<DimensionLayout dim="0">
<Group type="103" groupAlignment="0" attributes="0">

```

```

<Group type="102" alignment="1" attributes="0">
<EmptySpace pref="320" max="32767" attributes="0"/>
<Group type="103" groupAlignment="0" attributes="0">
<Component id="jLabel1" min="-2" pref="162" max="-2" attributes="0"/>
<Component id="jLabel2" min="-2" pref="104" max="-2" attributes="0"/>
</Group>
<EmptySpace min="-2" pref="240" max="-2" attributes="0"/>
</Group>
</Group>
</DimensionLayout>
<DimensionLayout dim="1">
<Group type="103" groupAlignment="0" attributes="0">
<Group type="102" alignment="1" attributes="0">
<EmptySpace pref="114" max="32767" attributes="0"/>
<Component id="jLabel2" min="-2" pref="46" max="-2" attributes="0"/>
<EmptySpace max="-2" attributes="0"/>
<Component id="jLabel1" min="-2" pref="84" max="-2" attributes="0"/>
<EmptySpace min="-2" pref="174" max="-2" attributes="0"/>
</Group>
</Group>
</DimensionLayout>
</Layout>
<SubComponents>
<Component class="javax.swing.JLabel" name="jLabel2">
<Properties>

```

```

<Property name="font" type="java.awt.Font"
editor="org.netbeans.beaninfo.editors.FontEditor">

<Font name="Showcard Gothic" size="18" style="0"/>

</Property>

<Property name="text" type="java.lang.String" value="WELCOME"/>

</Properties>

</Component>

<Component class="javax.swing.JLabel" name="jLabel1">

<Properties>

<Property name="font" type="java.awt.Font"
editor="org.netbeans.beaninfo.editors.FontEditor">

<Font name="Showcard Gothic" size="18" style="0"/>

</Property>

</Properties>

</Component>

</SubComponents>

</Container>

</SubComponents>

</Form>

<?xml version="1.0" encoding="UTF-8" ?>

<Form version="1.3" maxVersion="1.9"
type="org.netbeans.modules.form.forminfo.JFrameFormInfo">

<NonVisualComponents>

<Menu class="javax.swing.JMenuBar" name="jMenuBar1">

<SubComponents>

<Menu class="javax.swing.JMenu" name="jMenu1">

```

```

<Properties>

<Property name="text" type="java.lang.String" value="Home"/>

</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu1MouseClicked"/>

</Events>

</Menu>

<Menu class="javax.swing.JMenu" name="jMenu2">

<Properties>

<Property name="text" type="java.lang.String" value="New Image"/>

</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu2MouseClicked"/>

</Events>

</Menu>

<Menu class="javax.swing.JMenu" name="jMenu3">

<Properties>

<Property name="text" type="java.lang.String" value="Image Details"/>

</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu3MouseClicked"/>

</Events>

</Menu>

<Menu class="javax.swing.JMenu" name="jMenu4">

```

```

<Properties>

<Property name="text" type="java.lang.String" value="Send Details"/>

</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu4MouseClicked"/>

</Events>

</Menu>

<Menu class="javax.swing.JMenu" name="jMenu5">

<Properties>

<Property name="text" type="java.lang.String" value="Received Details"/>

</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu5MouseClicked"/>

</Events>

</Menu>

<Menu class="javax.swing.JMenu" name="jMenu6">

<Properties>

<Property name="text" type="java.lang.String" value="Logout"/>

</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent" handler="jMenu6MouseClicked"/>

</Events>

</Menu>

</SubComponents>

```

```

</Menu>

</NonVisualComponents>

<Properties>
<Property name="defaultCloseOperation" type="int" value="3"/>
</Properties>

<SyntheticProperties>
<SyntheticProperty name="menuBar" type="java.lang.String" value="jMenuBar1"/>
<SyntheticProperty name="formSizePolicy" type="int" value="1"/>
<SyntheticProperty name="generateCenter" type="boolean" value="false"/>
</SyntheticProperties>

<AuxValues>
<AuxValue name="FormSettings_autoResourcing" type="java.lang.Integer" value="0"/>
<AuxValue name="FormSettings_autoSetComponentName" type="java.lang.Boolean"
value="false"/>
<AuxValue name="FormSettings_generateFQN" type="java.lang.Boolean"
value="true"/>
<AuxValue name="FormSettings_generateMnemonicsCode" type="java.lang.Boolean"
value="false"/>
<AuxValue name="FormSettings_i18nAutoMode" type="java.lang.Boolean"
value="false"/>
<AuxValue name="FormSettings_layoutCodeTarget" type="java.lang.Integer"
value="1"/>
<AuxValue name="FormSettings_listenerGenerationStyle" type="java.lang.Integer"
value="0"/>
<AuxValue name="FormSettings_variablesLocal" type="java.lang.Boolean"
value="false"/>
<AuxValue name="FormSettings_variablesModifier" type="java.lang.Integer"
value="2"/>
</AuxValues>

```

```

<Layout>
<DimensionLayout dim="0">
<Group type="103" groupAlignment="0" attributes="0">
<Group type="102" alignment="0" attributes="0">
<EmptySpace max="-2" attributes="0"/>
<Component id="jPanel1" max="32767" attributes="0"/>
<EmptySpace max="-2" attributes="0"/>
</Group>
</Group>
</DimensionLayout>
<DimensionLayout dim="1">
<Group type="103" groupAlignment="0" attributes="0">
<Group type="102" alignment="0" attributes="0">
<EmptySpace max="-2" attributes="0"/>
<Component id="jPanel1" max="32767" attributes="0"/>
<EmptySpace max="-2" attributes="0"/>
</Group>
</Group>
</DimensionLayout>
</Layout>
<SubComponents>
<Container class="javax.swing.JPanel" name="jPanel1">

<Layout>
<DimensionLayout dim="0">

```

```

<Group type="103" groupAlignment="0" attributes="0">
<Group type="102" attributes="0">
<EmptySpace min="-2" pref="249" max="-2" attributes="0"/>
<Component id="jLabel1" min="-2" max="-2" attributes="0"/>
<EmptySpace max="32767" attributes="0"/>
</Group>
<Group type="102" alignment="0" attributes="0">
<EmptySpace min="-2" pref="66" max="-2" attributes="0"/>
<Component id="jButton2" min="-2" pref="180" max="-2" attributes="0"/>
<EmptySpace type="separate" max="-2" attributes="0"/>
<Component id="jButton1" min="-2" pref="180" max="-2" attributes="0"/>
<EmptySpace pref="29" max="32767" attributes="0"/>
<Component id="jButton3" min="-2" pref="180" max="-2" attributes="0"/>
<EmptySpace min="-2" pref="69" max="-2" attributes="0"/>
</Group>
<Group type="103" rootIndex="1" groupAlignment="0" attributes="0">
<Group type="102" attributes="0">
<EmptySpace min="-2" pref="65" max="-2" attributes="0"/>
<Component id="jScrollPane1" min="-2" pref="237" max="-2" attributes="0"/>
<EmptySpace type="unrelated" max="-2" attributes="0"/>
<Component id="jScrollPane2" min="-2" pref="344" max="-2" attributes="0"/>
<EmptySpace pref="66" max="32767" attributes="0"/>
</Group>
</Group>
</Group>
</DimensionLayout>

```

```

<DimensionLayout dim="1">
<Group type="103" groupAlignment="0" attributes="0">
<Group type="102" alignment="0" attributes="0">
<EmptySpace max="-2" attributes="0"/>
<Component id="jLabel1" min="-2" max="-2" attributes="0"/>
<EmptySpace pref="284" max="32767" attributes="0"/>
<Group type="103" groupAlignment="0" max="-2" attributes="0">
<Component id="jButton3" max="32767" attributes="0"/>
<Group type="103" groupAlignment="3" attributes="0">
<Component id="jButton2" alignment="3" min="-2" pref="43" max="-2" attributes="0"/>
<Component id="jButton1" alignment="3" min="-2" pref="43" max="-2" attributes="0"/>
</Group>
</Group>
<EmptySpace min="-2" pref="67" max="-2" attributes="0"/>
</Group>
<Group type="103" rootIndex="1" groupAlignment="0" attributes="0">
<Group type="102" attributes="0">
<EmptySpace min="-2" pref="64" max="-2" attributes="0"/>
<Group type="103" groupAlignment="0" max="-2" attributes="0">
<Component id="jScrollPane1" alignment="0" pref="0" max="32767" attributes="0"/>
<Component id="jScrollPane2" alignment="0" min="-2" pref="224" max="-2"
attributes="0"/>
</Group>
<EmptySpace pref="136" max="32767" attributes="0"/>
</Group>
</Group>

```

```

</Group>

</DimensionLayout>

</Layout>

<SubComponents>

<Component class="javax.swing.JLabel" name="jLabel1">

<Properties>

<Property name="font" type="java.awt.Font"
editor="org.netbeans.beaninfo.editors.FontEditor">

<Font name="Wide Latin" size="18" style="1"/>

</Property>

<Property name="text" type="java.lang.String" value="Upload Image"/>

</Properties>

</Component>

<Container class="javax.swing.JScrollPane" name="jScrollPane1">

<AuxValues>

<AuxValue name="autoScrollPane" type="java.lang.Boolean" value="true"/>

</AuxValues>

<Layout
class="org.netbeans.modules.form.compat2.layouts.support.JScrollPaneSupportLayout
"/>

<SubComponents>

<Component class="javax.swing.JTable" name="jTable_Files_Name">

<Properties>

<Property name="model" type="javax.swing.table.TableModel"
editor="org.netbeans.modules.form.editors2.TableModelEditor">

<Table columnCount="0" rowCount="0"/>

```

```

</Property>

<Property name="tableHeader" type="javax.swing.table.JTableHeader"
editor="org.netbeans.modules.form.editors2.JTableHeaderEditor">

<TableHeader reorderingAllowed="true" resizingAllowed="true"/>

</Property>

</Properties>

<Events>

<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent"
handler="jTable_Files_NameMouseClicked"/>

</Events>

</Component>

</SubComponents>

</Container>

<Container class="javax.swing.JScrollPane" name="jScrollPane2">

<AuxValues>

<AuxValue name="autoScrollPane" type="java.lang.Boolean" value="true"/>

</AuxValues>

<Layout
class="org.netbeans.modules.form.compat2.layouts.support.JScrollPaneSupportLayout
"/>

<SubComponents>

<Component class="javax.swing.JTable" name="jTable_Files_Name2">

<Properties>

<Property name="model" type="javax.swing.table.TableModel"
editor="org.netbeans.modules.form.editors2.TableModelEditor">

<Table columnCount="2" rowCount="0">

```

```

<Column editable="true" title="Details" type="java.lang.Object"/>
<Column editable="true" title="Values" type="java.lang.Object"/>
</Table>
</Property>
<Property name="tableHeader" type="javax.swing.table.JTableHeader"
editor="org.netbeans.modules.form.editors2.JTableHeaderEditor">
<TableHeader reorderingAllowed="true" resizingAllowed="true"/>
</Property>
</Properties>
<Events>
<EventHandler event="mouseClicked" listener="java.awt.event.MouseListener"
parameters="java.awt.event.MouseEvent"
handler="jTable_Files_Name2MouseClicked"/>
</Events>
</Component>
</SubComponents>
</Container>
<Component class="javax.swing.JButton" name="jButton2">
<Properties>
<Property name="text" type="java.lang.String" value="Download"/>
</Properties>
<Events>
<EventHandler event="actionPerformed" listener="java.awt.event.ActionListener"
parameters="java.awt.event.ActionEvent" handler="jButton2ActionPerformed"/>
</Events>
</Component>
<Component class="javax.swing.JButton" name="jButton1">

```

```

<Properties>

<Property name="text" type="java.lang.String" value="Original"/>

</Properties>

<Events>

<EventHandler event="actionPerformed" listener="java.awt.event.ActionListener"
parameters="java.awt.event.ActionEvent" handler="jButton1ActionPerformed"/>

</Events>

</Component>

<Component class="javax.swing.JButton" name="jButton3">

<Properties>

<Property name="text" type="java.lang.String" value="Dummy"/>

</Properties>

<Events>

<EventHandler event="actionPerformed" listener="java.awt.event.ActionListener"
parameters="java.awt.event.ActionEvent" handler="jButton3ActionPerformed"/>

</Events>

</Component>

</SubComponents>

</Container>

</SubComponents>

</Form>

<EmptySpace type="separate" max="-2" attributes="0"/>

```

6.3 SCREEN SHOTS

The screenshot displays a web application window titled "Location Transformation". Inside the window, there is a "REGISTRATION" form. The form is organized into two columns. The left column contains fields for "Name" (filled with "palani"), "Email" (filled with "palani@gmail.com"), and "Username" (filled with "palani"). The right column contains fields for "Contact" (filled with "9876543210"), "Address" (filled with "trichy"), and "Password" (filled with "3414"). Below these fields is a "Register" button. Below the main window, a "Message" dialog box is shown with the text "Registration success" and an "OK" button.

Name	Contact
palani	9876543210

Email	Address
palani@gmail.com	trichy

Username	Password
palani	3414

Register

Message
Registration success
OK

Figure 1. User Registration

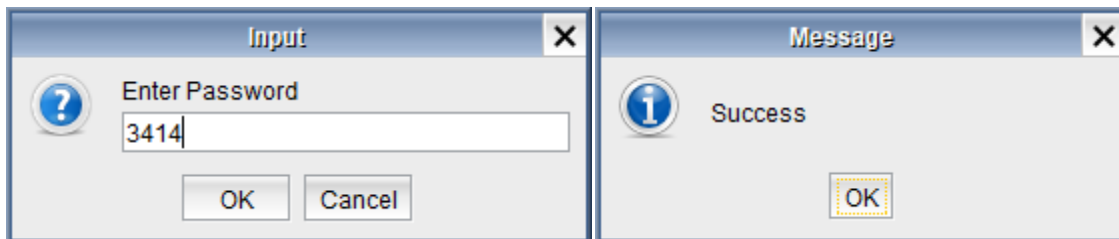


Figure 2. User Login success

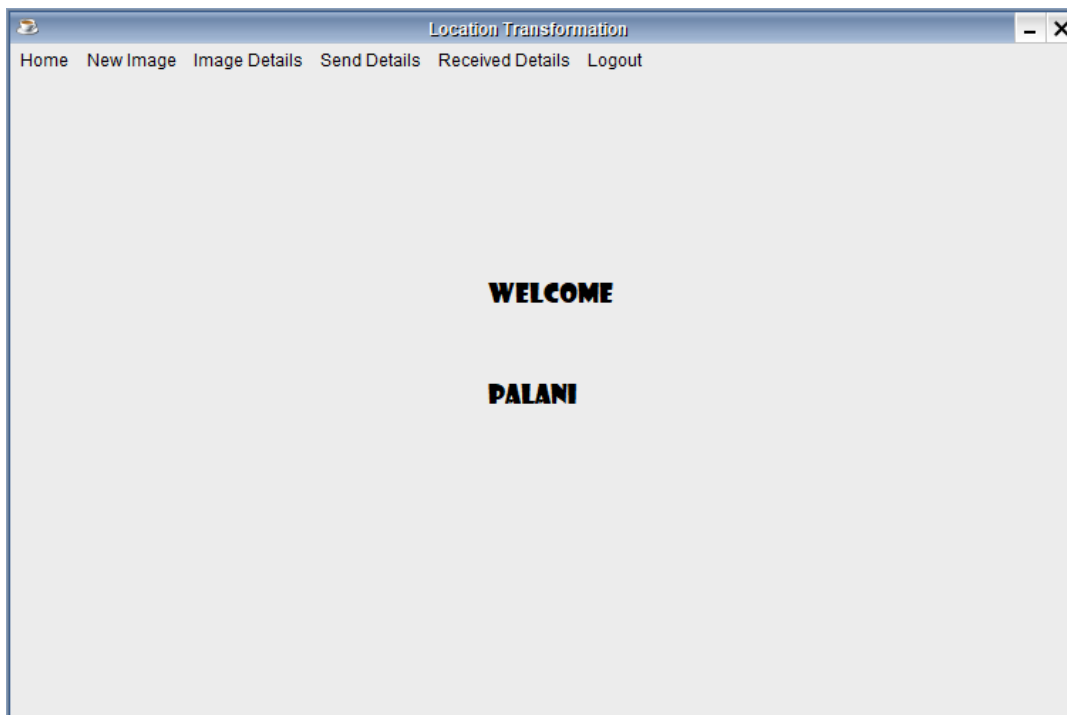


Figure 3. User Home Page

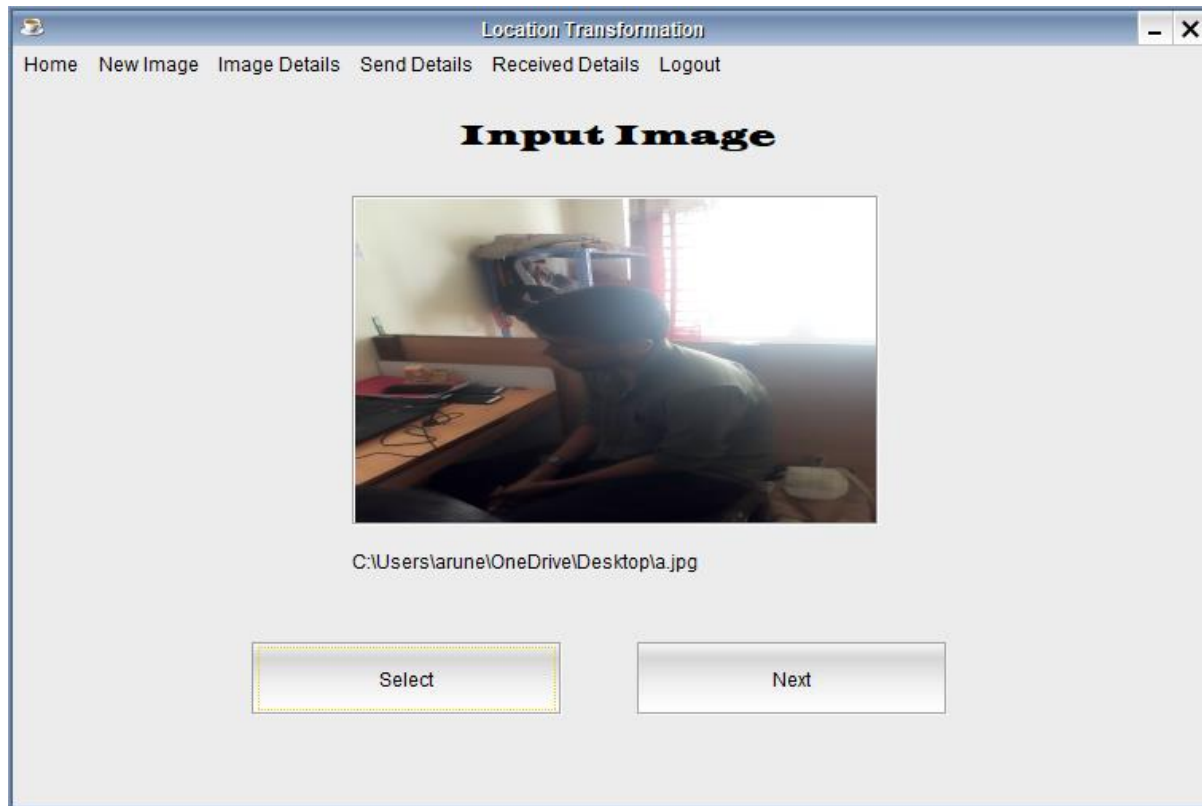
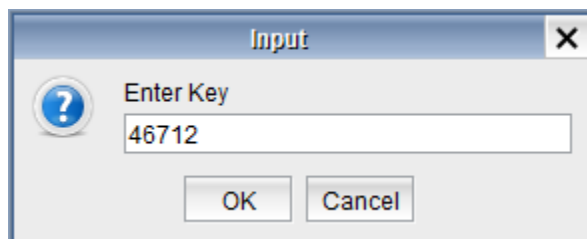
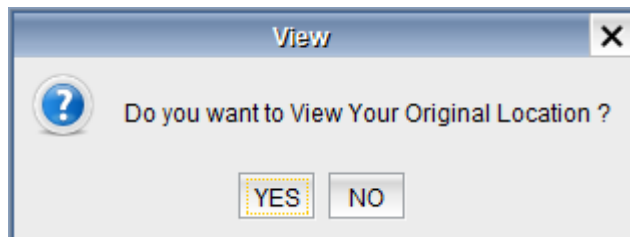


Figure 4. Upload Image



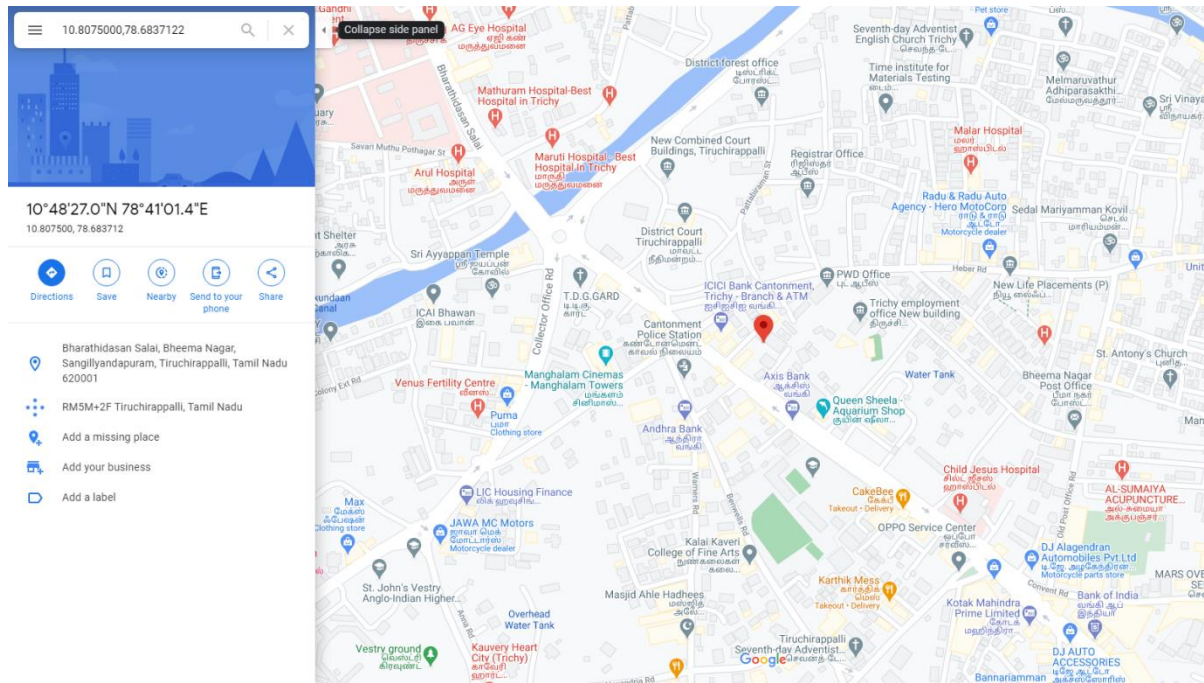


Figure 5. User View Original Location of Image

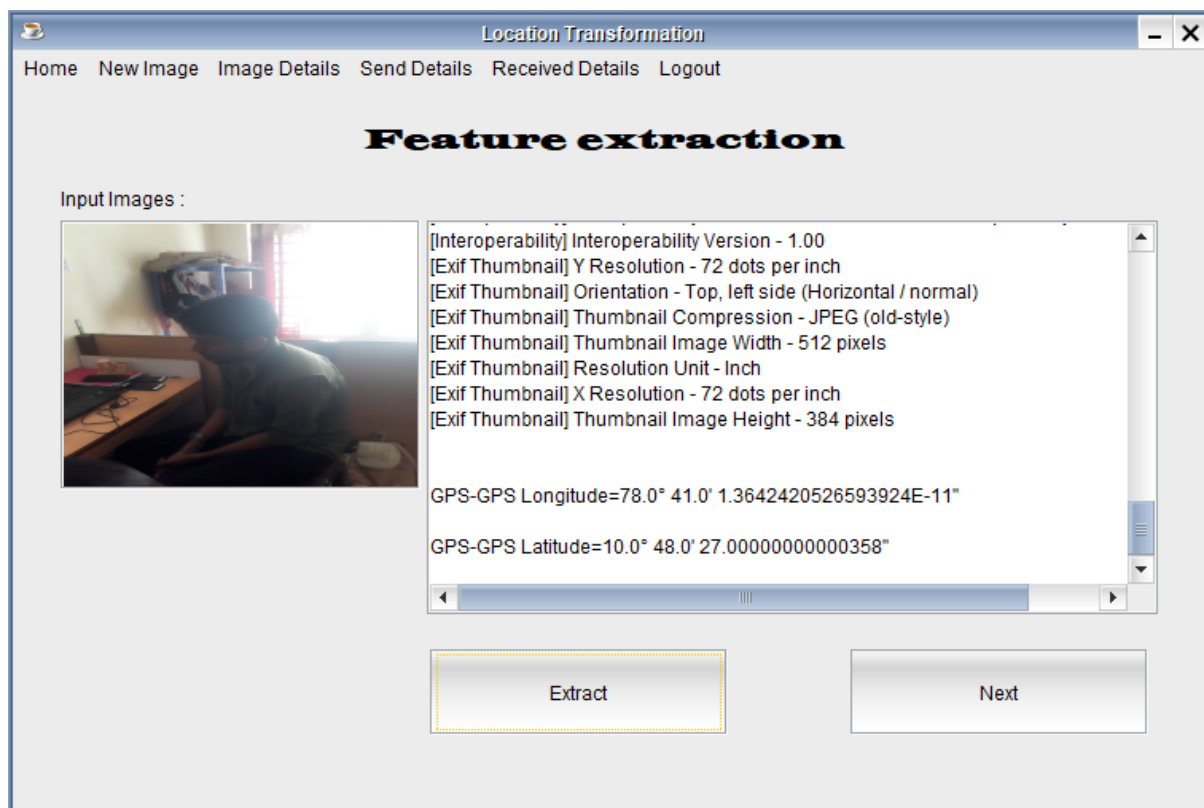


Figure 6. Feature Extraction

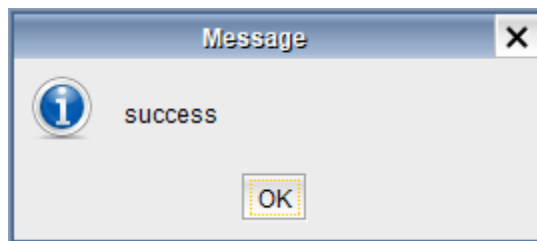
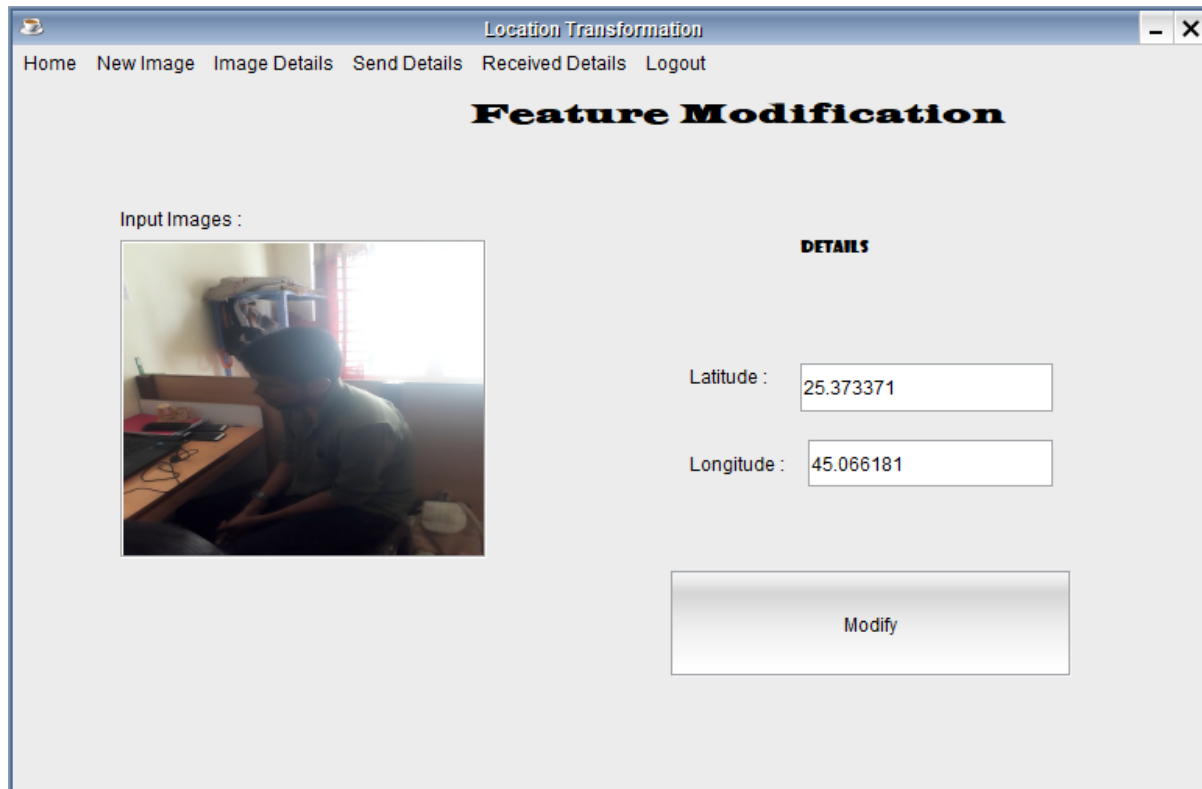
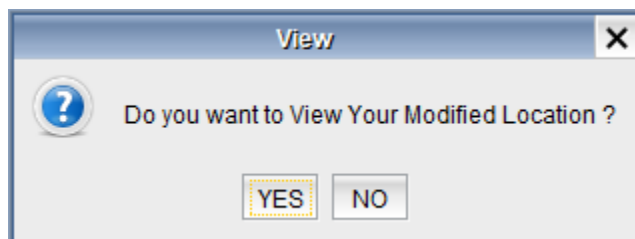


Fig 8.2.7 feature modification



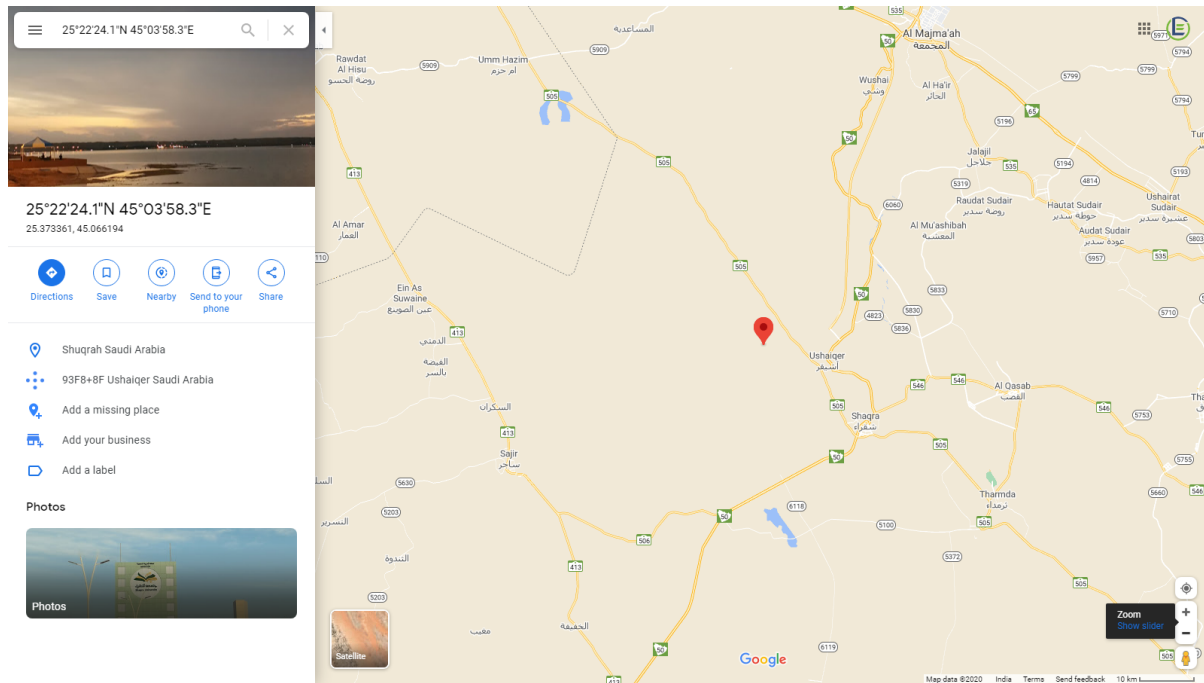


Figure 7. User View Modified Location of Image

Location Transformation

Home

New Image

Image Details

Send Details

Received Details

Logout

Feature Modification

Latitude 2 :

10.23223

Longitude 2 :

73.3434

Latitude 3 :

12.232321

Longitude 3 :

72.12312

Receiver :

palani

Random Key :

46712

Update

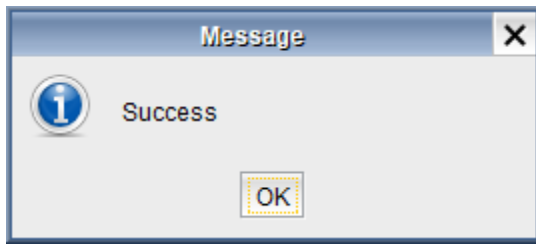


Figure 8. Feature Modification

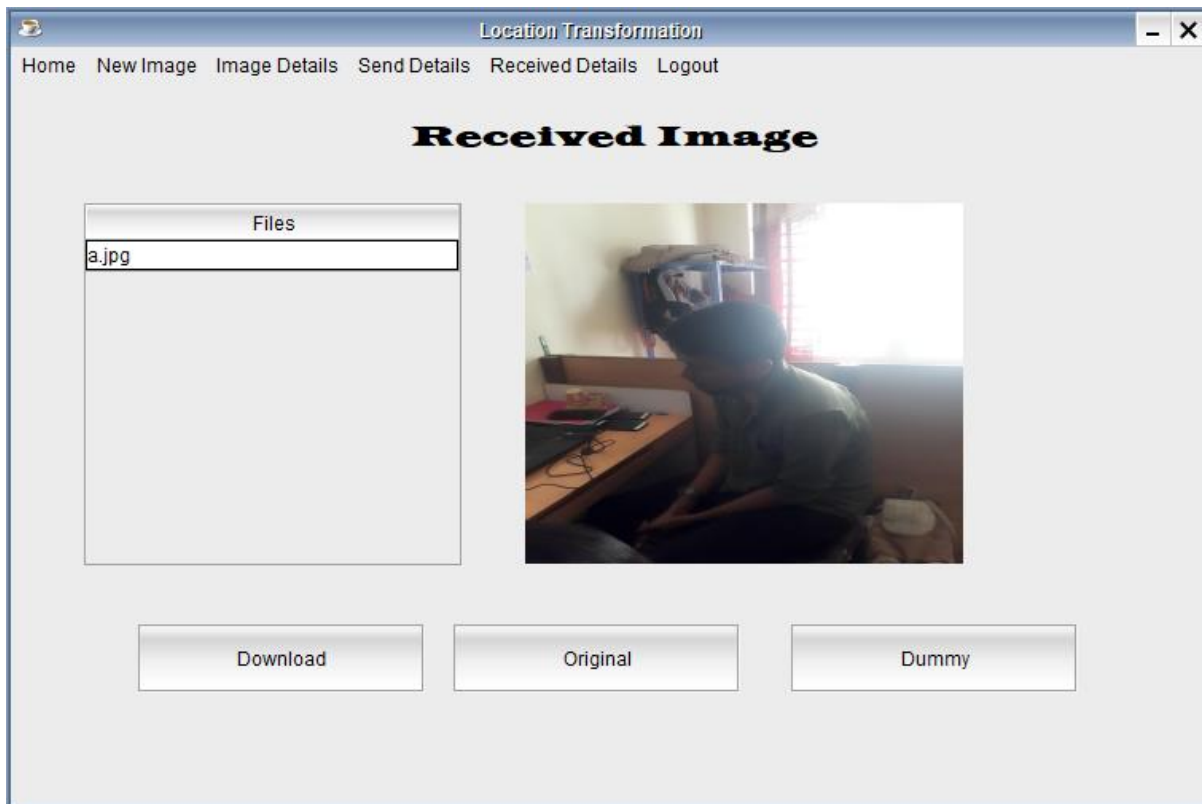


Figure 9. View Received Image

7. REFERENCES

Book Reference

- [1] Alvin J Bruney M V P, "The Microsoft Office The project b Components Black Book with .NET ", Lulu.com (9 April 2005)
- [2] Brennan Stehling, " Pro JAVA for SQL Server: High Performance Data Access for The project b Developers (Expert's Voice in SQL Server)", Apress; 1st ed. edition (26 September 2007)
- [3] Robin Dewson, " Beginning SQL Server 2005 Express for Developers: From Novice to Professional (Expert's Voice in .NET) ", Apress; 1st ed. edition (8 February 2007)
- [4] Matthew Macdonald, " JAVA: The Complete Reference", McGraw Hill Education (1 July 2017)

The project b reference

- [1] <https://dotnet.microsoft.com/>
- [2] https://en.wikipedia.org/wiki/.NET_Framework
- [3] <https://www.geeksforgeeks.org/introduction-to-net-framework/>