User Authentication Mechanisms on Android

M.Tech. Stage II Report

Submitted in partial fulfillment of the requirements for the degree of

Master of Technology

by

Anshita Agrawal
Roll No : 113050033

under the guidance of

Prof. D. B. Phatak

Department of Computer Science and Engineering
Indian Institute of Technology, Bombay

June 2013
Abstract

Authentication is the process of determining whether someone or something is, in fact, who or what it claim to be. The ways in which someone may be authenticated fall into three categories: something the user knows, something the user has, and something the user is. Authentication can be explicit, one time entry based or implicit and continuous. Implicit authentication based on behavioral biometrics is an authentication mechanism in which users are authenticated based on the way they interact with their mobile devices. Behavioral biometrics can include touch pattern, application usage pattern, typing pattern, device orientation pattern, walking pattern etc. In this research these behavioral biometric authentication techniques have been studied. These techniques work well for smart-phones as they provide desired accuracy for authenticating users.

With the increasing popularity of tablets and in view of the data and information that can be stored on the tablet, it is necessary to ensure the security of the data and information that is stored on the tablet. User authentication is an important security measure for protecting the information stored on the tablet because these devices have higher risk of theft. Also, an intruder may get hold of the device after initial authentication has been done. Hence, a continuous authentication method can either complement entry-point based authentication methods by monitoring the user after a successful login, or, if the method satisfies particular accuracy requirements, it could even substitute entry-point based authentication.
# Contents

1 Introduction  
   1.1 Authentication Mechanisms .................................................. 4  
      1.1.1 Passwords ........................................................................ 4  
      1.1.2 Certificates ..................................................................... 5  
      1.1.3 Smart Cards ..................................................................... 5  
      1.1.4 Biometrics ....................................................................... 5  
      1.1.5 Hardware Tokens ............................................................... 6  
      1.1.6 Proximity ......................................................................... 6  
   1.2 Biometric Authentication .......................................................... 6  
      1.2.1 Operating Modes ............................................................... 7  
      1.2.2 Performance Measurement ................................................. 8  
      1.2.3 Challenges ...................................................................... 9  

2 Literature Review  
   2.1 Face Recognition .................................................................. 10  
   2.2 Generic Framework ............................................................... 10  
      2.2.1 Face Detection ................................................................. 11  
      2.2.2 Feature Extraction ............................................................ 12  
      2.2.3 Face Classification ............................................................ 12  
   2.3 Face Recognition Techniques .................................................. 13  
   2.4 Comparison of Face Recognition Techniques .............................. 16  
   2.5 Challenges ........................................................................... 19  

3 Problem Formulation  
   3.1 Motivation ............................................................................ 20  
      3.1.1 Why implicit or continuous authentication .......................... 20  
      3.1.2 Why touchscreen devices and tablets ................................. 20  
   3.2 Problem Statement ................................................................. 21  

4 Implementation Details  
   4.1 Application Overview ............................................................. 23  
   4.2 Application Modules ............................................................... 23  
   4.3 Algorithms Used ................................................................. 25  
      4.3.1 Eigenface Approach ......................................................... 25  
      4.3.2 Fisherface ................................................................. 26  
   4.4 Threshold Selection ............................................................... 27  

5 Testing and Results .................................................................... 28
6 Conclusion and Future Work
# List of Figures

1.1 Basic Block Diagram of Biometric System[?] ........................................ 8
2.1 Generic Framework for Face Recognition [14] ........................................ 11
Chapter 1

Introduction

Authentication is the process of determining whether someone or something is, in fact, who or what it claim to be.

The ways in which someone may be authenticated fall into three categories, based on what are known as the factors of authentication: something the user knows, something the user has, and something the user is. Each authentication factor covers a range of elements used to authenticate or verify a person’s identity prior to being granted access, approving a transaction request, signing a document or other work product, granting authority to others, and establishing a chain of authority. The three factors (classes) and some of elements of each factor are:

- the ownership factors: Something the user has (e.g., wrist band, ID card, security token, software token, phone, or cell phone).
- the knowledge factors: Something the user knows (e.g., a password, pass phrase, or personal identification number (PIN), challenge response (the user must answer a question)).
- the inherence factors: Something the user is or does (e.g., fingerprint, retinal pattern, DNA sequence (there are assorted definitions of what is sufficient), signature, face, voice, unique bio-electric signals, or other biometric identifier).

An authentication mechanism may consist of one, two or all of the above mentioned factors and be therefore termed as One-factor authentication, Two-factor authentication or Three-factor authentication respectively.

1.1 Authentication Mechanisms

1.1.1 Passwords

Password is the most commonly one-factor authentication mechanism. It can be used in one of the following forms:

- Text based: Each user registers initially (or is registered by someone else), using an assigned or self-declared password. On each subsequent use, the user must know and use the previously declared password.
- Non text based: Require the user to tap symbols within a randomly generated matrix or a sequence of points on a photo.
1.1.2 Certificates

Certificates are electronic credentials that bind the identity of the certificate owner to a pair (public and private) of electronic keys that can be used to encrypt and sign information digitally. These electronic credentials assure that the keys actually belong to the person or organization specified. Each certificate contains at least the following information:

- Owner’s public key
- Owner’s name or alias
- Expiration date of the certificate
- Serial number of the certificate
- Name of the organization that issued the certificate
- Digital signature of the organization that issued the certificate

Certificates can also contain other user-supplied information, including a postal address, an e-mail address, and basic registration information, such as the country or region, postal code, age, and gender of the user.

1.1.3 Smart Cards

A smart card is a security chip, embedded in a credit card, badge or MMC/SD memory. That chip provides safe storage for cryptographic keys used by authentication and encryption algorithms. For example, a laptop may be unlocked by inserting an employee’s badge into the laptop’s card reader. When that employee launches a VPN tunnel or Wi-Fi connection, a certificate on the smart card can be automatically used for network authentication.

1.1.4 Biometrics

Biometric authentication covers fingerprints, voiceprints, iris scans, handwritten signatures, and so on. Enterprises have resisted biometrics because of cost, but some new business laptops and PDAs include fingerprint readers, and accept voice input. Biometrics are very convenient on frequently used mobile devices, but environment (e.g., dirt, noise) must also be considered.

Biometric systems are essentially pattern-recognition applications performing authentication using biometric attributes which can be classified into two categories:

- Physiological (like fingerprint, face, iris, retina, hand geometry, thermograms, DNA, earshape, body odour, vein patterns, electrocardiogram, brain waves etc.)
- Behavioural characteristics (like voice, signature, handwriting, gait, keystroke, lip motion etc.) that persons possess.
1.1.5 Hardware Tokens

A Hardware Token is an Authenticator in the form of a physical object, where the user’s interaction with a login system proves that the user physically possesses the object. Proving possession of the Token may involve one of several techniques:

- Reading a periodically changing pseudo-random number from the Token’s display and typing it into a login prompt.
- Keying a challenge string displayed by the login system into the Token, and typing a string that the Token displays as a result back into the login system.
- Plugging the Token into the workstation, using a USB port, or some other connection (parallel or serial port, smart card slot, etc.).

1.1.6 Proximity

A few mobile security products have started to support proximity-based authentication. A PDA or smartphone may stay unlocked indefinitely while communicating with the user’s Bluetooth headset. RFID tag readers are being used for proximity-based authentication, permitting connections with mobile devices that pass through a checkpoint and denying connections outside that area. Proximity authentication is not yet common but has the potential to provide more transparent mobile authentication in the future.

1.2 Biometric Authentication

Biometrics (or biometric authentication) refers to the identification of humans by their characteristics or traits. Biometric identifiers are often categorized as physiological versus behavioral characteristics. A physiological biometric would identify by one’s voice, DNA, hand print or behavior. Behavioral biometrics are related to the behavior of a person, including but not limited to: typing rhythm, gait, and voice.

Many different aspects of human physiology, chemistry or behavior can be used for biometric authentication. The selection of a particular biometric for use in a specific application involves a weighting of several factors. Seven such factors to be used when assessing the suitability of any trait for use in biometric authentication are as follows:

- **Universality** means that every person using a system should possess the trait.
- **Uniqueness** means the trait should be sufficiently different for individuals in the relevant population such that they can be distinguished from one another.
- **Permanence** relates to the manner in which a trait varies over time. More specifically, a trait with ‘good’ permanence will be reasonably invariant over time with respect to the specific matching algorithm.
- **Measurability** (collectability) relates to the ease of acquisition or measurement of the trait. In addition, acquired data should be in a form that permits subsequent processing and extraction of the relevant feature sets.
- **Performance** relates to the accuracy, speed, and robustness of technology used.
• **Acceptability** relates to how well individuals in the relevant population accept the technology such that they are willing to have their biometric trait captured and assessed.

• **Circumvention** relates to the ease with which a trait might be imitated using an artifact or substitute.

No single biometric will meet all the requirements of every possible application.

### 1.2.1 Operating Modes

A biometric system can operate in the following two modes:

1. **Verification Mode**: In verification mode the system performs a one-to-one comparison of a captured biometric with a specific template stored in a biometric database in order to verify the individual is the person they claim to be. Three steps involved in person verification. In the first step, reference models for all the users are generated and stored in the model database. In the second step, some samples are matched with reference models to generate the genuine and impostor scores and calculate the threshold. Third step is the testing step. This process may use a smart card, username or ID number (e.g. PIN) to indicate which template should be used for comparison. 'Positive recognition' is a common use of verification mode, "where the aim is to prevent multiple people from using same identity".

2. **Identification Mode**: In Identification mode the system performs a one-to-many comparison against a biometric database in attempt to establish the identity of an unknown individual. The system will succeed in identifying the individual if the comparison of the biometric sample to a template in the database falls within a previously set threshold. Identification mode can be used either for 'positive recognition' (so that the user does not have to provide any information about the template to be used) or for 'negative recognition' of the person "where the system establishes whether the person is who she (implicitly or explicitly) denies to be". The latter function can only be achieved through biometrics since other methods of personal recognition such as passwords, PINs or keys are ineffective.

The first time an individual uses a biometric system is called enrollment. During the enrollment, biometric information from an individual is captured and stored. In subsequent uses, biometric information is detected and compared with the information stored at the time of enrollment. Note that it is crucial that storage and retrieval of such systems themselves be secure if the biometric system is to be robust. The first block (sensor) is the interface between the real world and the system; it has to acquire all the necessary data. Most of the times it is an image acquisition system, but it can change according to the characteristics desired. The second block performs all the necessary pre-processing; it has to remove artifacts from the sensor, to enhance the input (e.g. removing background noise), to use some kind of normalization, etc. In the third block necessary features are extracted. This step is an important step as the correct features need to be extracted in the optimal way. A vector of numbers or an image with particular properties is used to create a template. A template is a synthesis of the relevant characteristics extracted from the source. Elements of the biometric measurement that are not used in the comparison algorithm are discarded in the template to reduce the filesize and to protect the identity of the enrollee.

If enrollment is being performed, the template is simply stored somewhere (on a card or within a database or both). If a matching phase is being performed, the obtained template is passed to a matcher that compares it with other existing templates, estimating the distance between them using
any algorithm (e.g. Hamming distance). The matching program will analyze the template with
the input. This will then be output for any specified use or purpose (e.g. entrance in a restricted
area). Selection of biometrics in any practical application depending upon the characteristic mea-
surements and user requirements. We should consider Performance, Acceptability, Circumvention,
Robustness, Population coverage, Size, Identity theft deterrence in selecting a particular biomet-
ric. Selection of biometric based on user requirement considers Sensor availability, Device availability,
Computational time and reliability, Cost, Sensor area and power consumption.

1.2.2 Performance Measurement

The critical promise of biometric recognition is that when a biometric sample is presented to the
system, it will offer the correct decision. However, a practical biometric system can make two basic
types of errors:

- False Match: the system incorrectly declares a successful match between the input pattern
  and a non-matching template in the database (in the case of identification) or a template
  associated with an incorrectly claimed identity (in the case of verification).

- False Non-match: the biometric system incorrectly declares failure of match between the input
  pattern and a matching pattern in the database (identification) or the pattern associated with
  the correctly claimed identity (verification).

The following are used as performance metrics for biometric systems:

1. False Acceptance Rate (FAR): the probability that the system incorrectly matches the input
   pattern to a non-matching template in the database. It measures the percent of invalid inputs
   which are incorrectly accepted. In case of similarity scale, if the person is imposter in real, but
   the matching score is higher than the threshold, then he is treated as genuine that increase
   the FAR and hence performance also depends upon the selection of threshold value.
2. False Rejection Rate (FRR): the probability that the system fails to detect a match between the input pattern and a matching template in the database. It measures the percent of valid inputs which are incorrectly rejected.

3. False Rejection Rate (FRR): the probability that the system fails to detect a match between the input pattern and a matching template in the database. It measures the percent of valid inputs which are incorrectly rejected.

1.2.3 Challenges

Although biometrics appears to be the obvious technology for robust person authentication, and has been successfully deployed in several niche markets, it is not yet a foolproof method of automatic human recognition. With the availability of inexpensive and compact biometric sensors and fast processing chips, it is becoming increasingly clear that a broader use of biometric technology would require better solutions to three fundamental barriers:

- Recognition performance: How to effectively represent and recognize biometric patterns (e.g., how to recognize a person with 99.999% accuracy)
- System security: How to guarantee that the biometric systems are not vulnerable to sabotage (e.g., can we ensure that fraudsters cannot infiltrate the system?)
- Privacy issues: How to make sure that the biometric system is being exclusively used for the expressed purpose (e.g., how to prevent trusted system administrators from abusing the system).
2.1 Face Recognition

A facial recognition device is one that views an image or video of a person and compares it to one that is in the database. It does this by comparing structure, shape and proportions of the face; distance between the eyes, nose, mouth and jaw; upper outlines of the eye sockets; the sides of the mouth; location of the nose and eyes; and the area surrounding the cheek bones. Upon enrolment in a facial recognition program, several pictures are taken of the subject at different angles and with different facial expressions. At time of verification and identification the subject stands in front of the camera for a few seconds, and then the image is compared to those that have been previously recorded.

The main facial recognition methods are: feature analysis, neural network, eigenfaces, automatic face processing. Some facial recognition software algorithms identify faces by extracting features from an image of a subject’s face. Other algorithms normalize a gallery of face images and then compress the face data, only saving the data in the image that can be used for facial recognition. A probe image is then compared with the face data.

The benefits of facial recognition are that it is non intrusive, can be done from a distance even without the user being aware they are being scanned. (i.e.: bank or government office) What sets apart facial recognition from other biometric techniques is that it can be used for surveillance purposes; as in searching for wanted criminals, suspected terrorists, and missing children. Facial recognition can be done from far away so with no contact with the subject so they are unaware they are being scanned. Facial recognition is most beneficial to use for facial authentication than for identification purposes, as it is too easy for someone to alter their face, features with a disguise or mask, etc. Environment is also a consideration as well as subject motion and focus on the camera. Facial recognition, when used in combination with another biometric method, can improve verification and identification results dramatically.

2.2 Generic Framework

In most cases, a face recognition algorithm can be divided into the following functional modules:

- a face image detector finds the locations of human faces from a normal picture against simple or complex background.
• a face recognizer determines who this person is.

Both the face detector and the face recognizer follow the same framework; they both have a feature extractor that transforms the pixels of the facial image into a useful vector representation, and a pattern recognizer that searches the database to find the best match to the incoming face image.

The difference between the two is the following: in the face detection scenario, the pattern recognizer categorizes the incoming feature vector to one of the two image classes: face images and non-face images. In the face recognition scenario, on the other hand, the recognizer classifies the feature vector (assuming it is from a face image) as As face, Bs face, or some other persons face that is already registered in the database. Figure 1 depicts one example of the face recognition system. Face detection is defined as the process of extracting faces from scenes. So, the system positively identifies a certain image region as a face. The next step feature extraction involves obtaining relevant facial features from the data. These features could be certain face regions, variations, angles or measures, which can be human relevant (e.g. eyes spacing) or not. This phase has other applications like facial feature tracking or emotion recognition. Finally, the system does recognize the face. In an identification task, the system would report an identity from a database. This phase involves a comparison method, a classification algorithm and an accuracy measure. These phases can be merged, or new ones could be added. Therefore, we could find many different engineering approaches to a face recognition problem. Face detection and recognition could be performed in tandem, or proceed to an expression analysis before normalizing the face.

2.2.1 Face Detection

Face detection algorithms usually share common steps. Firstly, some data dimension reduction is done, in order to achieve an admissible response time. Some pre-processing could also be done to adapt the input image to the algorithm prerequisites. Then, some algorithms analyze the image as it is, and some others try to extract certain relevant facial regions. The next phase usually involves extracting facial features or measurements. These will then be weighted, evaluated or compared to
decide if there is a face and where is it. Finally, some algorithms have a learning routine and they include new data to their models.

Face detection is, therefore, a two class problem where we have to decide if there is a face or not in a picture. This approach can be seen as a simplified face recognition problem. Face recognition has to classify a given face, and there are as many classes as candidates. Consequently, many face detection methods are very similar to face recognition algorithms. Or put another way, techniques used in face detection are often used in face recognition. Detection methods are divided into following categories:

- Knowledge-based methods: Ruled-based methods that encode our knowledge of human faces.
- Feature-invariant methods: Algorithms that try to find invariant features of a face despite its angle or position.
- Template matching methods: These algorithms compare input images with stored patterns of faces or features.
- Appearance-based methods: A template matching method whose pattern database is learnt from a set of training images.

These categories may overlap, so an algorithm could belong to two or more categories.

2.2.2 Feature Extraction

Feature extraction involves several steps - dimensionality reduction, feature extraction and feature selection. These steps may overlap, and dimensionality reduction could be seen as a consequence of the feature extraction and selection algorithms. Both algorithms could also be defined as cases of dimensionality reduction.

A feature extraction algorithm extracts features from the data. It creates those new features based on transformations or combinations of the original data. In other words, it transforms or combines the data in order to select a proper subspace in the original feature space. On the other hand, a feature selection algorithm selects the best subset of the input feature set. It discards non-relevant features. Feature selection is often performed after feature extraction. So, features are extracted from the face images, and then an optimum subset of these features is selected. The dimensionality reduction process can be embedded in some of these steps, or performed before them.

Dimensionality reduction is an essential task in any pattern recognition system. The performance of a classifier depends on the amount of sample images, number of features and classifier complexity. One could think that the false positive ratio of a classifier does not increase as the number of features increases. However, added features may degrade the performance of a classification algorithm. This may happen when the number of training samples is small relative to the number the features.

2.2.3 Face Classification

Once the features are extracted and selected, the next step is to classify the image. Appearance-based face recognition algorithms use a wide variety of classification methods. Sometimes two or more classifiers are combined to achieve better results. On the other hand, most model-based
algorithms match the samples with the model or template. Then, a learning method can be used to improve the algorithm. One way or another, classifiers have a big impact in face recognition. There are three key concepts in building a classifier - similarity, probability and decision boundaries.

1. **Similarity**: This approach is intuitive and simple. Patterns that are similar should belong to the same class. The idea is to establish a metric that defines similarity and a representation of the same-class samples. For example, the metric can be the Euclidean distance. The representation of a class can be the mean vector of all the patterns belonging to this class. The 1-NN decision rule can be used with these parameters. Its classification performance is usually good. This approach is similar to a k-means clustering algorithm in unsupervised learning. There are other techniques that can be used. For example, Vector Quantization, Learning Vector Quantization, Self-Organizing Maps etc. Other example of this approach is template matching. Researches classify face recognition algorithm based on different criteria. Some publications defined Template Matching as a kind or category of face recognition algorithms. However, we can see template matching just as another classification method, where unlabeled samples are compared to stored patterns.

2. **Probability**: Some classifiers are built based on a probabilistic approach. Bayes decision rule is often used. The rule can be modified to take into account different factors that could lead to miss-classification. Bayesian decision rules can give an optimal classifier, and the Bayes error can be the best criterion to evaluate features.

3. **Decision boundaries**: This approach can become equivalent to a Bayesian classifier. It depends on the chosen metric. The main idea behind this approach is to minimize a criterion (a measurement of error) between the candidate pattern and the testing pattern. One example is the Fishers Linear Discriminant. Its closely related to PCA. FLD attempts to model the difference between the classes of data, and can be used to minimize the mean square error or the mean absolute error. Other algorithms use neural networks. Multilayer perceptron is one of them. They allow nonlinear decision boundaries. However, neural networks can be trained in many ways, so they can lead to diverse classifiers. They can also provide a confidence in classification, which can give an approximation of the posterior probabilities. Assuming the use of a Euclidean distance criterion, the classifier could make use of the three classification concepts here explained.

### 2.3 Face Recognition Techniques

Face recognition is a process of identifying or verifying a person from an image and comparing the selected features from the image with a given database. The process of getting features from the face involves extraction of face from the rest of the image. The features such as distance between the eyes, width of the nose, shape of the cheekbones, and other distinguishable features are obtained. These features are then compared to the features obtained from the database to find a match. Following are the techniques used for face recognition:

- **Eigenfaces**: In this approach [12, 13], the face images are decomposed into a small set of characteristic feature images called eigenfaces from the original training set of images by means of principal component analysis (PCA). An important feature of PCA is that any original image can be reconstructed from the training set by a linear combination of the eigenfaces. Each eigenface represents only certain features of the face. However, the losses
due to omitting some of the eigenfaces can be minimized by choosing only the most important
features (eigenfaces). The eigenface approach involves the following initialization operations:

1. An initial set of images (training set) is acquired.
2. The eigenfaces from the training set are calculated and only M images that correspond
to the highest eigenvalues define the face space.
3. By projecting the face images onto the face space, the corresponding distribution in M
dimensional weight space for each individual image is found.

With these weights, any image in the database can be reconstructed using the weighted sum
of the eigenfaces. In order to recognize face images, the following steps are to be followed:

1. A set of weights based on the input image and the M eigenfaces are calculated by pro-
jecting the input image onto each of the eigenfaces.
2. Nearest neighbour classification is used in order to find out the unknown image in the
training set.

- **Fisherface**: The fisherface method of face recognition [11] uses both principal component
analysis (PCA) and linear discriminant analysis (LDA) to produce a subspace projection
matrix, similar to that used in the eigenface method. Differing from the Eigenface concept,
the fisherface method tries to maximize the ratio of the between-class scatter versus the
within-class scatter. The result of this shapes the projections so that the distances between
the classes are at a maximum, while the distances between samples of the same class are at
a minimum. A possible disadvantage is if the between-class scatter is large, then the within-
class scatter might also still be of a relatively large value. The fisherface approach is a widely
used method for feature extraction in face images.

- **Neural Networks**: The attractiveness of using neural networks could be due to its non
linearity in the network. The way by which a neural network structure is constructed is
crucial for successful recognition. It is very much dependent on the intended application. For
face detection, multilayer perceptron, convolutional neural network and self organizing maps
have been applied.
The MLP refer to the network consisting of a set of sensory units (source nodes) that constitute
the input layer, one or more hidden layers of computation nodes, and an output layer of
computation nodes. The input signal propagates through the network in a forward direction,
from left to right and on a layer-by-layer basis.
The SOM provides a quantization of the image samples into a topological space where inputs
that are nearby in the original space are also nearby in the output space, thereby providing
dimension reduction and invariance to minor changes in the image sample. The convolutional
network extracts successively larger features in a hierarchical set of layers and provides partial
invariance to translation, rotation, scale, and deformation.

- **Graph Matching**: In the Elastic Bunch Graph Matching algorithm (EBGM) [19], faces are
represented as labeled graphs, with nodes positioned at fiducial points (eyes, nose, mouth,
and etc.) based on Gabor Wavelet Transform (GWT). EBGM is claimed to be the best in
terms of identification rate and performance reliability. However, poor illumination reduces
recognition. A set of Gabor wavelet coefficients for each point is generated. Several feature
points representing the local features are extracted from the training faces. Each feature point is represented by a feature vector combining the GWT coefficients and its coordinates. All feature vectors are combined together to represent a face that is used in comparison and recognition process.

- **Hidden Markov Model**: Faces were intuitively divided into regions such as the eyes, nose, mouth, etc., which can be associated with the states of a hidden Markov model. Since HMMs require a one-dimensional observation sequence and images are two-dimensional, the images should be converted into either 1D temporal sequence or 1D spatial sequence. An unknown test image is first sampled to an observation sequence. Then, it is matched against every HMMs in the model face database (each HMM represents a different subject). The match with the highest likelihood is considered the best match and the relevant model reveals the identity of the test face.

- **Geometrical Feature Matching**: Geometrical feature matching techniques are based on the computation of a set of geometrical features from the picture of a face. The fact that face recognition is possible even at coarse resolution as low as 8x6 pixels [20] when the single facial features are hardly revealed in detail implies that the overall geometrical configuration of the face features is sufficient for recognition. The overall configuration can be described by a vector representing the position and size of the main facial features, such as eyes and eyebrows, nose, mouth, and the shape of face outline. Geometrical feature matching based on precisely measured distances between features may be most useful for finding possible matches in a large database.

- **Template Matching**: A simple version of template matching is that a test image represented as a two-dimensional array of intensity values is compared using a suitable metric, such as the Euclidean distance, with a single template representing the whole face. There are several other more sophisticated versions of template matching on face recognition. One can use more than one face template from different viewpoints to represent an individual’s face. The template matching is superior in recognition (100 percent recognition rate) One drawback of template matching is its computational complexity. Another problem lies in the description of these templates. Since the recognition system has to be tolerant to certain discrepancies between the template and the test image, this tolerance might average out the differences that make individual faces unique. In general, template-based approaches compared to feature matching are a more logical approach.

- **3D Morphable Model**: The morphable face model is based on a vector space representation of faces that is constructed such that any convex combination of shape and texture vectors of a set of examples describes a realistic human face. Fitting the 3D morphable model to images can be used in two ways for recognition across different viewing conditions [20]: First, after fitting the model, recognition can be based on model coefficients, which represent intrinsic shape and texture of faces, and are independent of the imaging conditions. Second, three-dimensional face reconstruction can also be employed to generate synthetic views from gallery probe images. The synthetic views are then transferred to a second, viewpoint-dependent recognition system.

- **Line Edge Map**: Edge information is a useful object representation feature that is insensitive to illumination changes to certain extent. Edge images of objects could be used for object recognition and to achieve similar accuracy as gray-level pictures.
A Line Edge Map approach [20] extracts lines from a face edge map as features. This approach can be considered as a combination of template matching and geometrical feature matching. The LEM approach not only possesses the advantages of feature-based approaches, such as invariance to illumination and low memory requirement, but also has the advantage of high recognition performance of template matching. Line Edge Map integrates the structural information with spatial information of a face image by grouping pixels of face edge map to line segments. After thinning the edge map, a polygonal line fitting process is applied to generate the LEM of a face.

- **Support Vector Machine:** SVM is a learning technique that is considered an effective method for general purpose pattern recognition because of its high generalization performance without the need to add other knowledge. Intuitively, given a set of points belonging to two classes, a SVM finds the hyperplane that separates the largest possible fraction of points of the same class on the same side, while maximizing the distance from either class to the hyperplane. This hyperplane is called Optimal Separating Hyperplane (OSH) which minimizes the risk of misclassifying not only the examples in the training set but also the unseen example of the test set. The main characteristics of SVMs are:

  1. They minimize a formally proven upper bound on the generalization error.
  2. They work on high-dimensional feature spaces by means of a dual formulation in terms of kernels.
  3. The prediction is based on hyperplanes in these feature spaces, which may correspond to quite involved classification criteria on the input data.
  4. The outliers in the training data set can be handled by means of soft margins.

- **Multiple Classifier System:** Recently, Multiple Classifier Systems (MCSs) based on the combination of outputs of a set of different classifiers have been proposed in the field of face recognition as a method of developing high performance classification systems. Traditionally, the approach used in the design of pattern recognition systems has been to experimentally compare the performance of several classifiers in order to select the best one. However, an alternative approach based on combining multiple classifiers has emerged over recent years and represented a departure from the traditional strategy. This approach goes under various names such as MCS or committee or ensemble of classifiers, and has been developed to address the practical problem of designing automatic pattern recognition systems with improved accuracy.

### 2.4 Comparison of Face Recognition Techniques

Comparison of various face recognition techniques has been done by several people. But so far no technique exists which has shown satisfactory results under all circumstances. A comparative study of various techniques based on results obtained by various groups of people is shown below. In [6] feature graphs based on a wavelet transform, principle component analysis (PCA), and linear discriminant analysis (LDA) are compared. They reported 88%, 85% and 56% accuracy for PCA, LDA, and Gabor Wavelets respectively with a database containing 20 and individuals varying in gender, age, pose, and race. For each individual five images were used for testing, while one images was employed as the learning sample. Efficiency of face recognition tested on positive and negative samples was reported best for PCA followed by LDA. Gabor wavelets performed worst.
In [12, 13] the authors compared Eigenface and Fisherface approach by size of training data and by image pose. They tested both algorithms on 20 images for varying number of poses in training data. The recognition for fisherface turned out to be better than eigenface approach when the number of poses were less. But as the number increased, the percentage of true recognition in both cases was almost same. They also tested both algorithms at their optimum working conditions on various poses of same image. At their optimum trained conditions the recognition was almost same.

The authors of [20] reported 96%, 85%, and 64% correct classifications averaged over lighting, orientation, and size variations, respectively for eigenfaces. Their database contained 2,500 images of 16 individuals. 96.2% correct recognition was reported for Neural networks approach on ORL database of 400 images of 40 individuals. The classification time is less than 0.5 second, but the training time is as long as 4 hours.

The Elastic Bunch Graph Matching (EBGM) process is computationally expensive, taking about 25 seconds to compare with 87 stored objects on a parallel machine with 23 transputers. They reported recognition rates of 86.5% and 66.4% for the matching tests of 111 faces of 15 degree rotation and 110 faces of 30 degree rotation to a gallery of 112 neutral frontal views. In general, dynamic link architecture is superior to other face recognition techniques in terms of rotation invariance; however, the matching process is computationally expensive.

The recognition rate of Hidden Markov Model (HMM) approach is 87% using ORL database consisting of 400 images of 40 individuals. A pseudo 2D HMM was reported to achieve a 95% recognition rate in their preliminary experiments.

Feature based approach extracts a set of geometrical features from the picture of a face, such as nose width and length, mouth position, and chin shape. In [20] there were 35 features extracted from a 35 dimensional vector. The recognition was then performed with a Bayes classifier. They reported a recognition rate of 90% on a database of 47 people.

Another approach used Gabor wavelet decomposition to detect feature points for each face image which greatly reduced the storage requirement for the database. Typically, 35-45 feature points per face were generated. The recognition accuracy in terms of the best match to the right person was 86% and 94% of the correct person’s faces were in the top three candidate matches.

The percentage of correct identification of 3D Morphable approach on CMU-PIE database, based on side-view gallery, was 95% and the corresponding percentage on the FERET set, based on frontal view gallery images, along with the estimated head poses obtained from fitting, was 95.9%.

Line Edge Map (LEM) makes use of edge maps to measure the similarity of face images. 92% accuracy was achieved. Experiments on frontal faces under controlled /ideal conditions indicate that LEM is consistently superior to edge map. LEM correctly identify 100% and 96.43% of the input frontal faces on face databases, respectively.

In [17], a comparative study of three appearance based face recognition methods namely PCA, LDA and Independent Component Analysis (ICA) and their accompanied three distance metrics (Manhattan, Euclidean and Cosine) in equal working conditions was performed. LDA with cosine metric outperformed PCA and ICA. For all probe sets the cosine metric seems to be the best choice.
for LDA and ICA and Manhattan for PCA.

In [4], three algorithms namely PCA, LDA and Bayesian Intrapersonal/Extrapersonal Classifier (BIC) are compared. PCA algorithm outperformed the other algorithms in terms of accuracy, efficiency, productivity and robustness. Results proved PCA produced an average recognition rate of 93%, while BIC lagged behind with an average recognition rate of 83%. Although BIC was the least efficient in terms of speed, LDA was the least accurate in terms of identification, as it scored an average recognition rate of 80%.

The authors of [19] analyzed the performance of PCA, LDA and EBGM algorithms from three points of view: recognition accuracy rate, computational cost, and tolerance. The results show that both LDA and EBGM perform better than PCA. PCA runs fast but the accuracy rate and recognition tolerance is relatively low. EBGM is time-consuming, tends to have the highest accuracy rate and better recognition tolerance. LDA has both good accuracy rate and small computational time but a weaker tolerance than EBGM.

In [7], PCA, EBGM and interpersonal image difference classifier (IIDC) algorithms are examined. The generalized conclusion drawn from the results shown is that PCA performs better than EBGM and IIDC algorithms.

Eigenface and Fisherface find face space based on the common face features of the training set images. Elastic Bunch Graph Matching take local face features like eye, mouth into account for recognition. According to [11] the accuracy of Eigenface is satisfactory (over 90%) with frontal faces. Eigenface uses PCA. A drawback is that it is very sensitive for lightening conditions and the position of the head.

Fisherface is similar to Eigenface but with improvement in better classification of different classes image. It uses both PCA and LDA. It could have better accuracy in facial expression than Eigenface approach. It is more invariant to light intensity. Fisherface is more complex than Eigenface and requires a lot of processing time. Besides, due to the need of better classification, the dimension of projection in face space is not as compact as Eigenface, resulting in larger storage of the face and more processing time in recognition.

EBGM has the advantage that changes in one feature (e.g. eyes open, closed) do not necessarily mean that the person is not recognized any more. This algorithm makes it possible to recognise faces up to a rotation of 22 degrees. Drawbacks of this algorithm are that it is very sensitive to lightening conditions and that a lot of graphs have to be placed manually on the face.

In authors of [15] have compared PCA, LDA, HMM, BIC algorithms. Comparing the results from each of the algorithms with respect to pose angle variance, LDA ranks first, followed by PCA, close behind is BIC with HMM being the last. For illumination angle variance, LDA performs the best, followed by BIC, PCA and HMM. Thus, LDA performs best with respect to changes in both pose angle and illumination angle.

In [18] PCA, LDA, ICA and Support Vector Machine (SVM) have been employed on AT&T and IFD databases to study their performance in terms of accuracy, training time, total execution time and model size. The following results were observed. SVM performed best in terms of accuracy for
AT&T database and LDA performed best for IFD database. ICA consumes more computational
time than other algorithms. PCA, LDA, and ICA have same testing time for both databases. The
model size of face image is small in PCA and LDA as compared to others. SVM has the largest
model size.

The authors of [16] compared Eigenfaces approach with Line Edge Map approach. Results show
that LEM allows better results for lighting and size variations. It beats eigenfaces method with size
variation. On the other hand, eigenfaces algorithm demonstrated better results for posing changes
than LEM.

The experiments in [2] suggests that both Fisherface and Eigenface methods perform well if
presented with an image in the test set which is similar to an image in the training set. The Fisher-
face method appeared to be the best at simultaneously handling variation in lighting and expression.

The above results suggest that Eigenface and Fisherface algorithms are the most promising
amongst all the approaches used for face recognition. It is difficult to say which among the two
approaches is better as both approaches performs well under different scenarios. Elastic Bunch
Graph Mapping algorithm provides the highest accuracy rate, but it takes significantly more time
for training and testing. Hence, in spite of providing highest accuracy rate this algorithm is not
the most efficient. All these algorithms have been tested on standard databases made available by
various people but not on real time images. Therefore, we have compared Fisherface and Eigenface
algorithms in real time scenario using the FaceApp application for doing continuous user authenti-
cation.

2.5 Challenges

There are several challenges associated with face and facial feature detection and can be attributed
by the following factors [1].

- Intensity: There are three types of intensity- color, gray, and binary.
- Pose: Face images vary due to the relative camera-face pose and some facial features such as
  an eye may become partially or fully occluded.
- Structural components: Facial features such as beards, mustaches, and glasses may or may
  not be presented.
- Image rotation: Face images directly vary by different rotations.
- Poor quality: Image intensity in poor-quality images, such as blurred images, distorted images,
  and images with noise.
- Facial expression: The appearance of faces depends on a personal facial expression.
- Occlusion: Faces may be partially occluded by other objects such as hand, scarf, etc.
- Illumination: Face images may vary due to the position of light source.
Chapter 3

Problem Formulation

3.1 Motivation

3.1.1 Why implicit or continuous authentication

Most methods for authenticating users on desktop computers or mobile devices define an entry point into the system. Typically, the user faces a password challenge and is granted access only if she inputs the correct password. While such entry-point based methods dominate the authentication schemes today, they have flaws from both, usability and security perspectives. From a usability perspective, traditional authentication schemes are inconvenient because users must focus on the authentication step every time they begin interacting with their device. Such inconvenience is amplified under the usage pattern of mobile devices, since they are more frequently accessed, and each use is typically shorter. Authentication with a PIN or secret gesture is too cumbersome for short bursts of activity, such as briefly checking one's email or reading an SMS. Hence, users often choose simple and weak secrets (or increase the screen lock timeouts of their devices) which makes the entry-point based authentication methods also flawed from a security standpoint. Furthermore, the device cannot detect intruders after the authentication step is performed successfully.

Mobile devices are at a higher risk of loss or theft compared to desktop computers. Continuous or implicit authentication approaches would provide an additional line of defense, designed as a non-intrusive and passive security countermeasure. Such approaches monitor the users interaction with the device, and ideally, at every point in time (or at least with a high frequency) the system estimates if the legitimate user is using the device. Hence, a continuous authentication method can either complement entry-point based authentication methods by monitoring the user after a successful login, or, if the method satisfies particular accuracy requirements, it could even substitute entry-point based authentication.

3.1.2 Why touchscreen devices and tablets

Although there is a growing body of literature about keystroke dynamics or mouse dynamics for continuous authentication, there is surprisingly little work on continuous authentication for touchscreen devices. The growing popularity of mobile devices and tablets increases the value of research on their security mechanisms. Specifically, to the best of our knowledge, there is no existing method for continuous authentication based on touch biometrics (i.e., without requiring a dedicated activity of the user). One reason might be the difficulty of extracting a set of sufficiently discriminative features from touch data, because users atomic navigation behavior mostly consists of simple and
In this research, continuous authentication scheme that relies on user’s real time image as a data source, has been developed. The scheme performs continuous authentication by capturing user’s image using device’s camera and matching it against the template created while training the face recognizer. Our goal was to analyze how robustly such schemes could operate and if they were sufficiently reliable to be used on tablets.

There are differences between smart phones and tablet computers that might make it harder to continuously authenticate users on tablets. In particular, the small size of the screen of smart phones helps continuous authentication. The reason is that content of documents, emails, image collections, menus, or icon collections hardly fit on the smart phones screen in most application scenarios. As a result, the user must move around screen content and thus the classifier gets a lot of observations over time. In contrast, on large tablet screens users can read for a long time without scrolling, all icons fit on screen, and so on. This might reduce the strokes per minute below a rate that can be considered secure. Moreover, the large screen introduces more degrees of freedom.

### 3.2 Problem Statement

**to be edited** Behavioural authentication mechanisms for user authentication based on touch biometrics are in use today for authenticating smartphone users. Continuous or implicit authentication is required to ensure that an intruder does not get access to the mobile device once the initial authentication challenge is passed. With the increasing popularity of tablets and in view of the confidential information that can be stored on a tablet it is necessary to ensure that the data or information stored in the tablet is secure. Implicit behavioural authentication provides sufficient accuracy so that it can be deployed for use on smartphones. As the touchscreen size of tablet is much larger than that of smartphone we need to investigate if it is possible to authenticate tablet users based on user’s interaction with the touchscreen. The goal is to analyze if these behavioural biometrics are sufficiently reliable to be used on tablets.

In particular, the questions to be answered are:

- Does the classification framework used on smartphones can be used for authentication on tablets?
- How user interaction with the touchscreen will vary because of the large screen size?
- What is the probability of rejecting a legitimate user?
- What is the probability of accepting an attacker?
- How long does the classifier need to make an authentication decision?
- What are the factors that affect the user’s behaviour?
- How the classification framework can be improved for increasing the accuracy of authentication mechanism?

The objective of this work is:

- To analyze whether the classification framework used for authenticating smartphone users based on touch biometrics can be used on tablets.
• To identify the feature set that can be used for authenticating tablet users with sufficient accuracy.

• To explore behavioural biometrics that can be used for implicit user authentication over tablets.

• To devise an application to test the feasibility and accuracy of the proposed authentication mechanism.
Chapter 4

Implementation Details

4.1 Application Overview

The FaceApp application performs continuous authentication based on Face Recognition technique to authenticate the user of the Aakash tablet. The application asks the user to first train the face recognizer by capturing certain number of images. Then it asks the user to enter a password which will be used when the recognizer reports negatively about the current user i.e. the face of the user trying to use the tablet doesn’t match with the training face. Once this is done, the application asks the user to select the mode of usage that the user wants. It has two modes. First, lock the selected applications installed on your tablet and second, lock the whole tablet. Whenever the user tries to use the tablet or one of the selected applications, the FaceApp checks whether the trained face matches with the face of the user currently using the tablet or not. If the face doesn’t match the application will lock the device or the selected application depending upon the mode selected. The user will then have to enter a password to be able to use the device again. The application performs recognition automatically after the intended duration of time preventing the illegitimate user from using the device even after device has passed the initial unlock challenge.

4.2 Application Modules

The FaceApp application has the following modules:

1. **Enable FaceApp Service:** This module enables/disables FaceApp service. While the service is active the application will perform continuous authentication of the tablet’s user by capturing user’s image after the specified interval of time. The user must enable this service in order to be able to perform continuous authentication using FaceApp.

2. **Auto Start Service:** This module lets the users choose whether they want the FaceApp service running after the device reboots. If this option is disabled the user will have to manually start the FaceApp service after the device reboots. It is recommended to keep this option enabled otherwise the security of the device could be compromised on device reboot.

3. **Select Mode:** The application has two modes of operation. The user can choose to operate the application in any of the two modes. Following are the modes:
• Lock selected applications: In this mode, the user can select the application which he/she wants to lock using FaceApp from all the applications that are installed on his/her tablet. Continuous authentication will be performed only while using the selected applications. The rest of the applications will be accessible to any user.

• Lock complete device: In this mode, continuous authentication will be performed irrespective of what application the user is using. This mode is useful if the user needs very high security for his/her device.

4. **Train Recognizer:** This module trains the face recognizer for recognizing the legitimate user of the tablet by capturing certain number of images of user’s face in real time scenario. The user can capture any but greater than a specified number of images to train the recognizer. The images should be captured in different lightings, face expressions and environments. This will help in recognizing the legitimate user more accurately.

5. **Face Detection:** This module has been merged with the training and authenticating modules. When the user’s face is captured either for training the recognizer or for authenticating the user, face detection will be done first. The application will capture an image only after a face has been detected by the same.

6. **Application Selection:** This module lets the user select the application which he/she wishes to lock using the FaceApp application. All the applications which are installed on user’s device will be displayed. The user can select any or all of those applications. After selecting the applications continuous authentication will be performed while using those applications, if the FaceApp service is enabled and recognizer has been trained.

7. **Set Password:** Password is provided as an alternative to unlock the device or selected applications when the face of the current user doesn’t match with the training data. This is required because the face recognition algorithms are not 100 percent accurate and could reject the legitimate user. Therefore, an alternative has been provided using which the user can use the device or selected application.

8. **Relock Policy:** Relock policy enables the user to select a relock timeout. Relock timeout is the time duration after which the user wish to lock the applications that he/she has selected from the select apps tab. For example, if the user has put a lock on Gmail application then whenever he/she will start gmail the FaceApp will capture his/her face image and unlock the application only if the user’s face matches with the training faces. When the gmail application gets unlocked the FaceApp will capture user’s face again after relock timeout duration and perform the same check. This will be repeated till the user stops using the gmail application.

9. **Authentication:** Authentication is performed continuously by capturing the face image of the person using the device currently. Once the face image is captured that image is matched with the training data. The user will be allowed to perform the intended task only if the captured image matches with the training data. Otherwise the user will be prompted to enter a password without which the user won’t be able to use the device or selected application. This will be repeated after every, say, 10 minutes while the user is using the device or the locked application.
4.3 Algorithms Used

4.3.1 Eigenface Approach

In this approach, the face images are decomposed into a small set of characteristic feature images called eigenfaces (which contain the common features in a face) which are extracted from the original training set of images by means of principal component analysis. An important feature of PCA is that any original image can be reconstructed from the training set by a linear combination of the eigenfaces. Each eigenface represents only certain features of the face. However, the losses due to omitting some of the eigenfaces can be minimized by choosing only the most important features (eigenfaces) [12].

The eigenface approach involves the following initialization operations:

1. An initial set of images (training set) is acquired.
2. The eigenfaces from the training set are calculated and only M images that correspond to the highest eigenvalues define the face space.
3. By projecting the face images onto the face space, the corresponding distribution in M-dimensional weight space for each individual image is found.

With these weights, any image in the database can be reconstructed using the weighted sum of the eigenfaces.

In order to recognize face images, the following steps are to be followed:

1. A set of weights based on the input image and the M eigenfaces are calculated by projecting the input image onto each of the eigenfaces.
2. Nearest neighbor classification is used in order to find out the unknown image in the training set.

Initialization

Let the training set of face images be $T_1, T_2, T_3, \ldots, T_M$. This training data set has to be mean adjusted before calculating the covariance matrix or eigenvectors. The average face is calculated as $\Psi = (1/M) \sum_{1}^{M} T_i$.

Each image in the data set differs from the average face by the vector $\phi = T_i \Psi$. This is actually mean adjusted data. The covariance matrix is

\begin{equation}
C = (1/M) \sum_{1}^{M} \phi_i \phi_i^T = AA^T
\end{equation}

where $A = [\phi_1, \phi_2, \ldots, \phi_M]$. The matrix $C$ is a $N^2$ by $N^2$ matrix and would generate $N^2$ eigenvectors and eigenvalues. With image sizes like 256 by 256, or even lower than that, such a calculation would be impractical to implement.

A computationally feasible method was suggested to find out the eigenvectors. If the number of images in the training set is less than the no of pixels in an image (i.e $M < N^2$), then we can solve an $M$ by $M$ matrix instead of solving a $N^2$ by $N^2$ matrix. Consider the covariance matrix as $A^T A$ instead of $AA^T$. Now the eigenvector $v_i$ can calculated as follows,

\begin{equation}
A^T A v_i = \mu_i v
\end{equation}
where $\mu_i$ is the eigenvalue. Here the size of covariance matrix would be $M$ by $M$. Thus we can have $m$ eigenvectors instead of $N^2$. Premultiplying equation 2 by $A$, we have

$$3) AA^T Av_i = \mu_i Av_i$$

The right hand side gives us the $M$ eigenfaces of the order $N^2$ by 1. All such vectors would make the image space of dimensionality $M$. The $M$ eigenfaces which have the largest associated eigenvalues are selected. These eigenfaces now span a $M$-dimensional subspace instead of $N^2$.

**Recognition**

A new image $T$ is transformed into its eigenface components (projected into face space) by a simple operation,

$$4) w_k = u_k^T(T - \psi)$$

where $k = 1,2,...,M$. The weights obtained as above form a vector $\Omega^T = [w_1, w_2, w_3,...,w_M]$ that describes the contribution of each eigenface in representing the input face image. The Euclidean distance of the weight vector of the new image from the face class weight vector can be calculated as follows,

$$5) \varepsilon_k = \| \Omega - \Omega_k \|$$

where $\Omega_k$ is a vector describing the $k^{th}$ face class. The face is classified as belonging to class $k$ when the distance $\varepsilon_k$ is minimum.

### 4.3.2 Fisherface

The fisherface approach is a widely used method for feature extraction in face images. This approach tries to find the projection direction in which, images belonging to different classes are separated maximally. Mathematically, it tries to find the projection matrix (the weights) in such a way that the ratio of the between-class scatter matrix and the in-class scatter matrix of projected images is maximized [12].

For a 'c' class problem, (where $c$ is number of individuals) the between class scatter matrix is

$$S_b = \sum_{i=1}^{c} Pr(\Omega_i)(\mu_i - \mu)(\mu_i - \mu)^T$$

The within class scatter matrix is given by :

$$S_w = \sum_{i=1}^{c}(Pr(\Omega_i) \times (1/N_i) \sum_{y_k \epsilon \Omega_i} (y_k - \mu_i)(y_k - \mu_i)^T)$$

for the images to be projected in such a way that they are maximal separated is solved by eigen value equation

$$S_b W = S_w W A$$

once the weights are obtained, they are used to project the images into the face space. After the weight basis is obtained, the recognition process is the same as in the case of eigenface algorithm.
4.4 Threshold Selection

Threshold selection is important because the threshold value decides whether the incoming image belongs to the person in the training set or not. Consider for simplicity we have only 10 images in the training set. And an image that is not in the training set comes up for the recognition task. The score for each of the 10 images will be found out with the incoming image. In addition, even if an image is not in the database, it will still say the image is recognized as the training image with which its score is the lowest. Clearly, this is incorrect. It is for this purpose that we decide the threshold.

Generally, threshold value is chosen arbitrarily. There is no formula for calculating the threshold value. Its value is chosen arbitrarily or taken as some factor of maximum value of minimum Euclidian distances of each image from other images [8]. In this algorithm the threshold value ranges from minimum value of minimum Euclidian distances of each image from other images to maximum value of minimum Euclidian distances of each image from other images.
Chapter 5

Testing and Results

to be added
Chapter 6

Conclusion and Future Work

to be edited

Authentication mechanisms for user authentication based on single entry point and continuous authentication have been studied. The three classes of authentication something the user has, something the user is, and something the user does have been examined. The authentication mechanisms one-way, two-way and three-way on Android devices have been studied. The ways of combining biometrics with cryptography to improve the security and to maintain the privacy and integrity of biometric traits have been analyzed. Behavioural authentication mechanisms and challenges in implementing them on Android devices are explored.

Authentication schemes (explicit and implicit) based on passwords, screen unlock patterns, fingerprint, gestures, cryptography, mobile usage patterns, orientation of the device, user interaction with the touchscreen etc have been studied. Also, how the combination of two or more schemes can improve the security of authentication schemes was investigated. Implicit or continuous authentication method can either complement entry-point based authentication methods by monitoring the user after a successful login, or, if the method satisfies particular accuracy requirements, it could even substitute entry-point based authentication.

Behavioural biometric authentication mechanisms for authenticating users based on their interaction with the mobile devices have been studied. A classification framework has been analyzed for classifying the feature set extracted from the user interaction with the touchscreen of the mobile device to investigate if it is possible to authenticate users while they perform basic navigation steps on a touchscreen device and without any dedicated and explicit security action that requires attention from the user.

Analysis and investigation of how robustly such schemes will operate will be done and if they are sufficiently reliable to be used on tablets will be tested. An application to test the feasibility and accuracy of the proposed approach will be devised. Feature set for improving the efficiency and accuracy of the classification framework will be explored.
Acknowledgment

It’s a great pleasure and moment of immense satisfaction for me to express my profound gratitude to my respected supervisor, Prof. D. B. Phatak whose constant encouragement enabled me to work enthusiastically. His perpetual motivation, patience and excellent expertise in discussion progress of the project work have benefited me to an extent, which is beyond commendation. His insight of knowledge of this domain made me realize that theoretical knowledge always helps to develop efficient operational software, which is a blend of all core subjects of the field. I am immensely indebted to him for his invaluable guidance and ever-ready support in the successful completion of stage-I of this project in time. Working under his guidance has been a fruitful and unforgettable experience. I express gratitude for his encouragement and for providing and outstanding academic environment for working in the department. I am thankful to him for providing the adequate facilities. I extend our sincere thanks to all the faculty members and the staff members of the Computer Science & Engineering Department, for their cooperation. Special thanks to my colleagues and friends for providing me useful comments, suggestion and continuous encouragement.
Bibliography


