CHAPTER 1

INTRODUCTION

1.1 PREAMBLE

Multimodal biometric authentication systems were proven to be efficient and given better security and accuracy in many online transaction systems and applications when compared to conventional security procedures. Multimodal biometrics were applied to various authentication and recognition services where high security is a major concern. But keeping the privacy of enrolledsubjects is the biggest challenge for biometric security researchers. To solve the challenges in privacy preservations of enrolled subjects, numerous template privacy protection schemes were proposed. Features generated from the Multimodal biometrics samples were protected by converting them into an unintelligible format using encryption techniques so that hackers were not able to compromise the Multimodal biometrics template easily. This way of protecting the enrolled subjects was called a template protection scheme.

1.2 OVERVIEW OF THE PROJECT

This template protection is an important factor because there are many hazards of identity theft and privacyleak issues, which are enrolled in databases secured by biometric credentials and raise privacy concerns. The goal is to increase the Exhaustive search and employ biometrics in a way that protects privacy while maximizing user control, minimizing the possibility of abuse, and ensuring the flawless operation of systems that use biometrics. A person's facial image should not be stored in a database directly. Biometric recognition using biometric encryption uses features likeface, finger print, finger vein to encrypt (code) extra data just the biometrically encrypted data is kept, not any other information, such as a cryptographic key. But due to the rapid growth in attacking knowledge among the adversaries, simple encryption techniques were not sufficient to give complete security protecting the templates.

As a result of the advancement of the Internet and Artificial Intelligence technology, in manyapplication domains, the method based on biometric identification verification is

displacing the conventional password-based access control and identity authentication technology. For Internet applications, secure identity identification is crucial. The widely used Biometric technique isnothing but Multimodal Biometric recognition. Particularly, it seems that facial recognition technologies have grown in popularity and are being used for entry and border control, as well as monitoring public locations.

In past years it might seem that there were many issues raised about personal information leaks and Hacking. The corruption of data is a global issue. The data in the database is not always safe. Various biometric template protection schemes were already proposed using face, Finger Prints, Iris. To protect that data, we should perform some Privacy preserving Techniques. This is because our data like bank information, personal documents, Health monitoring data whatever maybe should provide security Schemes .Various biometric template protection schemes were already proposed using face. Some of the methods applied CNN, entropy, and hashing for templateprotection in some cases those systems failed in indexing and Time computation issues. The recognition efficiency was quite the same for every proposed method but lag in Exhaustive SearchEfficiency. Unfortunately, it was observed from the literature that the approaches proposed were not supporting indexing, which would provide increased exhaustive search efficiency and reduced computation complexity in various template protection schemes. Also, it is identified from the results of those schemes that the hash-based indexing methods were given better performance whencompared to conventional approaches, especially privacy-preserving face identification systems that use a hashing look-up table to index and retrieve protected face templates. Additionally, this paper is intended to analyze the high level of privacy protection for the enrolled subjects which was ensured by the homomorphic encryption algorithms used to safeguard these Multimodal Biometric templates and summarizes various template protection schemes, encryption algorithms used to protect the templates, and concealment techniques.

Here the analyzed the security enhancements and exhaustive search capability of the methods proposed in the current context and it is identified that indexing will reduce the time complexity and enhance the exhaustive search.

The rest of this paper's chapters were organized withbackground study, related

work,and discussions.

1.2.1.1 Challenges in Template protection scheme

Template Protection Scheme for Secure the Multimodal Biometricsusing hash generation unique challenges.

- The most challenging problem is to protect the plain biometrics received during the enrolment time.
- Some papers explained that proposed system needs huge storage requirementsto store the features of faces and other biometrics.
- In some papers there is vulnerable recognition of faces and biometrics. In somemethods proposed method have to follow potential privacy issues to store the data. When an attacker aware of the biometric symmetry of two or more biometric templates used in amulti -biometrics.
- The Exhaustive search and Recognition Accuracy must and should high.

1.3 BACKGROUND STUDY OF TEMPLATE PROTECTION

Template Protection Scheme (TPS) is a security mechanism used to safeguard the personalinformation of an individual in biometric systems. It involves generating a template (a mathematicalrepresentation of biometric data) that is protected by encryption and other security measures. This template is used to verify the identity of an individual during subsequent biometric authentication.

One approach to implementing TPS for multimodal biometrics is through the use of hash generation. In this method, the biometric data is hashed (converted into a fixed-length string of characters) using a cryptographic hash function. The resulting hash value is then used to create thetemplate.

To ensure the security of the template, a secret key is used to encrypt the hash value before storingit. The encrypted hash value can only be decrypted using the secret key, which is kept secureandknown only to authorized parties. During authentication, the biometric data is hashed using thesame cryptographic hash function, and the resulting hash value is encrypted using the secret key. The encrypted hash value is then compared to the encrypted hash value stored in the template. If thetwo values match, the individual is authenticated.

This approach to TPS provides a high level of security for multimodal biometric systems, as the original biometric data is never stored and the template is protected by

encryption and a secret key.It also allows for efficient storage and processing of biometric data, as the fixed-length hash value is much smaller than the original data.

Overall, the use of hash generation in TPS is a promising approach to securing multimodal biometric systems and protecting the personal information of individuals. However, it is importantto ensure that appropriate cryptographic techniques are used to prevent attacks such as collision attacks and rainbow table attacks.

Then these images were selected for training, testing, and assessing the effectiveness of machinelearning and artificial intelligence (AI) algorithms, also known as common vision algorithms,which are included in an image dataset. Features were extracted by some feature- extracting methods like CNN, Vector, etc. Now the features get Embedded were converted into unintelligible format, which means a method of encryption. By using a secret key, picture encryption transforms a plain image into an encrypted one. CNN is used as the following step in the data acquisition process. Layers of a deep convolutional neural network. Convolutional and pooling layers are usually switched back and forth. Convolutional neural networks with deep learning.

1.2.1 Methods for Template Protection

There are several methods that can be used for template protection scheme (TPS) to safeguard biometric data. Some of these methods are:

Cryptographic Hashing: This is a widely used method for TPS in which biometric data is hashed (converted into a fixed-length string of characters) using a cryptographic hash function. The resulting hash value is then used to create the template. To ensure the security of the template, a secret key is used to encrypt the hash value before storing it.

Fuzzy Extractors: Fuzzy extractors are another commonly used method for TPS that involves generating a secret key from biometric data. This secret key is used to encrypt the template, whichcan only be decrypted using the same key. Fuzzy extractors can handle noisy biometric data by extracting a stable and consistent key from multiple noisy biometric inputs.

Cancelable Biometrics: This method involves transforming the original biometric

data into a new,cancelable form that cannot be reverse-engineered to recover the original biometric information. This transformed data is used to generate the template, which is protected by encryption and other security measures.

Random Projections: In this method, the original biometric data is projected onto a random subspace, and the resulting projection is used to create the template. The projection process is designed to be reversible, allowing the original biometric data to be recovered if necessary.

Biometric Encryption: This method involves encrypting the original biometric data using a publickey encryption scheme. The encrypted data is used to generate the template, which is stored in encrypted form. The decryption key is kept secret and is used to decrypt the template during authentication. Each of these methods has its own strengths and weaknesses, and the choice of method will depend on the specific requirements of the biometric system and the level of security needed.

1.3.1 Steps to protect the template

Feature Extraction: Feature extraction is the process of transforming raw data into a set of meaningful and informative features that can be used for analysis or modeling. In many fields, including computer vision, speech recognition, and natural language processing, feature extractionis a critical step in building machine learning models.

Cryptographic Hashing: This is a widely used method for TPS in which biometric data is hashed (converted into a fixed-length string of characters) using a cryptographic hash function. The resulting hash value is then used to create the template. To ensure the security of the template, a secret key is used to encrypt the hash value before storing it

Fuzzy Extractors: Fuzzy extractors are another commonly used method for TPS that involves generating asecret key from biometric data. This secret key is used to encrypt the template, whichcan only be decrypted using the same key. Fuzzy extractors can handle noisy biometric data by extracting a stable and consistent key from multiple noisy biometric inputs.

Cancelable Biometrics: This method involves transforming the original biometric

data into a new,cancelable form that cannot be reverse-engineered to recover the original biometric information. This transformed data is used to generate the template, which is protected by encryption and other security measures.

Random Projections: In this method, the original biometric data is projected onto a random subspace, and the resulting projection is used to create the template. The projection process is designed to be reversible, allowing the original biometric data to be recovered if necessary.

Biometric Encryption: This method involves encrypting the original biometric data using a publickey encryption scheme. The encrypted data is used to generate the template, which is stored in encrypted form. The decryption key is kept secret and is used to decrypt the template during authentication.

Each of these methods has its own strengths and weaknesses, and the choice of method willdepend on thespecific requirements of the biometric system and the level of security needed, vision, speech recognition, and natural language processing, feature extraction is a critical step in buildingmachine learning models.

The goal of feature extraction is to extract the most relevant and informative features from raw data while reducing the dimensionality of the data. This is done by identifying patterns and relationships in the data and transforming them into a set of features that can be easily processed bya machine learning algorithm.

In computer vision, for example, feature extraction is often used to extract features such asedges, corners, and textures from images. These features can be used to recognize objects, classifyimages, or perform other tasks. In speech recognition, features such as the Mel frequency cepstralcoefficients (MFCCs) are extracted from audio signals to represent the spectral characteristics of Speech sounds.

Homomorphic encryption: Homomorphic encryption is a form of encryption that allows computations to be performed on ciphertext, without the need to decrypt the data first. This means that data can remain encrypted throughout the entirecomputation process, preserving privacy and security.

In traditional encryption schemes, data is encrypted before transmission to prevent

unauthorized access, but it must be decrypted before it can be processed. Homomorphic encryption,on the other hand, allows computations to be performed directly on encrypted data, without requiring decryption.

There are two main types of homomorphic encryption:

Fully Homomorphic Encryption (FHE): FHE is the most general type of homomorphic encryption and allows any computation to be performed on encrypted data, including complex algorithms such as machine Learning and artificial intelligence. However, FHE is computationally intensive and is not yet practical for most applications. Partially Homomorphic Encryption (PHE):

PHE allows only certain types of computations to be performed on encrypted data, such as additionor multiplication. PHE is more practical than FHE and has been used in several applications, such as secure computation of financial data.

Homomorphic encryption has several advantages, including:

Privacy: Homomorphic encryption allows data to remain encrypted throughout the entirecomputation process, providing strong privacy guarantees.

Security: Homomorphic encryption protects against attacks that exploit weaknesses in the decryptionprocess.

Efficiency: While FHE is still computationally intensive, PHE can be more efficient than other securecomputation methods.

Homomorphic encryption is an active area of research, and while it is not yet widely used in practice,it has the potential to revolutionize the way sensitive data is processed and analyzed.

Hash key generation: Hash key generation is the process of generating a fixed-length, unique stringof characters from a given input data. Hash keys are commonly used in cryptography, digital signatures, and password storage, amongother applications. The process of generating hash keys is typically done using a hash function. A hash function is a mathematical function that takes an input message of arbitrary length and generates a fixed- length output, called a hash value or message digest. The output is usually a string of hexadecimal digits that serves as a unique identifier for the input data.

The key characteristics of a good hash function include:

Deterministic: A hash function should produce the same hash value for the same input every time itis run.

Uniformity: A good hash function should produce uniformly distributed hash values, with no biastowardsany particular input.

Non-invertibility: It should be computationally infeasible to reconstruct the original input datafrom the hash value.

Collision resistance: A good hash function should produce different hash values for differentinputs, and should make it difficult to find two different inputs that produce the same hash value.

There are many hash functions available, including MD5, SHA-1, SHA-2, and SHA-3. Thechoice of hash function will depend on the specific application and the level of security required.

Matching rate: Matching rate refers to the accuracy or success rate of a biometric authentication system in matching a given biometric sample (such as a fingerprint, face image, or voice recording) to a

stored template or database of biometric templates. The matching rate is typically expressed as a percentage or a ratio of the number of successful matches to the total number of comparison attempts. For example, if a system successfully matches 90 out of 100 attempts, the matching rate would be 90%.

The matching rate of a biometric authentication system depends on several factors, including: **Quality of the biometric sample**: The quality of the biometric sample used for comparison affectsthe matching rate. If the sample is noisy or of poor quality, the system may not be able to accuratelymatch it to the stored template.

Size and diversity of the biometric database: The size and diversity of the biometric database canaffect the matching rate. A larger and more diverse database may improve the system's ability to match a wider range of biometric samples.

Type of biometric technology: Different biometric technologies have different

matching rates. For example, iris recognition and facial recognition are generally considered more accurate than fingerprint recognition.

Threshold setting: The threshold setting for the system determines the level of similarity requiredfor a match to be considered successful. A higher threshold may result in a lower matching rate buta higher level of security.

The matching rate is an important metric for evaluating the performance of a biometric authentication system. High matching rates are important for ensuring the accuracy and reliability of the system, while low matching rates can lead to false positives or false negatives, which can compromise security.

1.3.2 Applications of Template Protection Scheme

Template protection schemes (TPS) are used to protect the privacy and security of biometric templates and are typically used in biometric authentication systems. Here are some common applications of TPS:

Mobile biometric authentication: TPS can be used to secure biometric authentication systems onmobile devices, such as smartphones and tablets. This can help to protect against unauthorized access the fraud.

Financial services: TPS can be used to secure biometric authentication systems used in financial services, such as banking and payment systems. This can help to protect against fraud and improvethe overall security of financial transactions.

Healthcare: Biometric authentication systems are increasingly being used in healthcare applications, such as patient identification and electronic health records. TPS can help to protect the privacy and security of biometric templates used in these systems.Overall, TPS can be applied in a wide range of biometric authentication systems to protect the privacy and security of biometrictemplates. They play a crucial role in ensuring the accuracy and reliability of biometric authentication systems while maintaining the privacy and security of sensitive data.

1.4 PROBLEM STATEMENT

Biometric systems typically function in two modes: identification and verification,

depending on the application context. A biometric claim is verified using a one-to-one biometric comparison process called biometric verification. Protecting the plain templates is the key elementin any template protection schemes. The face templates are secured using completely homomorphicencryption techniques which ensures the enrolled subjects high level of privacy protection.

1.5 AIM OF THE PROJECT

The aim of a Template Protection Scheme (TPS) project is to develop a secure and efficient method for protecting biometric templates used in biometric authentication systems. The project typically involves designing and implementing a TPS algorithm that can protect biometric templateswhile maintaining high levels of accuracy and reliability in the authentication process.

1.6 OBJECTIVE OF THE PROJECT

The objectives of a Template Protection Scheme (TPS) project can vary depending on thespecific application and context, but some common objectives may include:

Developing a secure and efficient TPS algorithm: The primary objective of a TPS project is to develop a robust and secure algorithm that can protect biometric templates from unauthorized access or misuse. The algorithm should be efficient enough to handle large-scale authentication systems and provide a high level of accuracy in matching biometric samples with the protected templates.

Evaluating the performance of the TPS algorithm: Another objective of a TPS project is to evaluate the performance of the TPS algorithm and compare it with existing methods for template protection. The evaluation may involve testing the algorithm on a large dataset of biometric templates and samples to assess its accuracy, efficiency, and robustness.

Ensuring compliance with privacy and security standards: TPS projects should ensure compliancewith privacy and security standards, such as the General Data Protection Regulation (GDPR) andISO/IEC 24745, which provide guidelines for the protection of biometric data. The TPSalgorithmshould be designed to meet these standards and ensure that biometric templates are protected against unauthorized access or misuse.

Developing a user-friendly interface: TPS projects may also involve developing

a user-friendly interface for the biometric authentication system that incorporates the TPS algorithm. The interface should be intuitive and easy to use, providing a seamless authentication experience for users while maintaining the security and privacy of their biometric data.

Ensuring interoperability and scalability: TPS projects should ensure that the developed TPS algorithm can be integrated into existing biometric authentication systems and be scalable enough tohandle large-scale authentication systems. Interoperability and scalability are essential for the adoption and widespread use of TPS in various applications.

Overall, the objectives of a TPS project are to develop a secure, efficient, and user-friendly methodforprotecting biometric templates used in biometric authentication systems while ensuringcompliance with privacy and security standards.

1.7 PROPOSED METHODOLOGY

The proposed methodology for Template Protection Scheme (TPS) can vary depending on thespecific application and context, but some common steps involved in a TPS project may include: **Biometric data acquisition:** The first step in a TPS project is to acquire the biometric data, such asfingerprints, facial images, or iris scans, from the users who will be enrolled in the system. Thebiometric data must be collected according to established standards and protocols to ensure accuracyand reliability.

Biometric feature extraction: The next step is to extract the relevant features from the biometric datathat will be used for authentication. This involves identifying unique patterns and characteristics in thebiometric data that can be used to distinguish one user from another.

Template creation and storage: Once the biometric features are extracted, a biometric template is created for each user. The template is a mathematical representation of the user's biometric data that canbe stored in a secure database for future use.

Template protection: The TPS algorithm is then applied to the biometric template to protect it from unauthorized access or misuse. The TPS algorithm may use techniques such as homomorphic encryption, fuzzy commitment, or secure sketch to protect the template.

Authentication: When a user attempts to authenticate, their biometric sample is compared to the protected template using a matching algorithm. The matching algorithm should be designed to provide accurate and reliable authentication while maintaining the privacy and security of the biometricdata.

Evaluation and optimization: The TPS algorithm is evaluated and optimized to ensure that it meets thedesired performance metrics, such as accuracy, efficiency, and security. The algorithm may be tested ona large dataset of biometric templates and samples to assess its performance and identify areas for improvement.

Integration and deployment: Once the TPS algorithm is optimized, it can be integrated into the biometric authentication system and deployed for use in real-world applications. The TPS algorithm should be designed to ensure interoperability and scalability, allowing it to be used in awide range of applications and contexts.

Overall, the proposed methodology for TPS involves a series of steps, from biometric data acquisition to authentication, aimed at protecting biometric templates while maintaining the accuracy and reliabilityof the authentication system. The methodology should be designed to ensure compliance with privacy and security standards and provide a user-friendly interface for seamless authentication.

1.8 SIGNIFICANCE OF THE WORK

TPS is significant because it enhances privacy and security, improves accuracy and reliability, ensures compliance with regulations, is interoperable and scalable, and provides a better user experiencefor biometric authentication systems.

1.9 LIMITATION OF THE WORK

- There is high complexity with this project.
- The effectiveness and efficiency can be improved by using Hashing which makes out the existing model even faster.

1.10 ORGANISATION OF THE REPORT

The remainder of the chapter is laid out as follows.

Chapter 1: This chapter provides an introduction, problem statement, project goal, methodology, worksignificance, and conclusion.

Chapter 2: This chapter offers a review of the literature as well as a comparison of severalstrategies for segmenting and classifying brain tumors.

Chapter 3: This chapter provides System Requirements such as Hardware Tools, Communication Interfaces, and Software/Hardware Requirements.

Chapter 4: It consists of System design, including System Architecture Chapter5: Itexplains how the project will be implemented and how it will be approached.

Chapter 6: Experimental results, project outcomes, and comparisons to existing methodologies are all included in this chapter.

Chapter 7: contains the project's conclusion and future enhancements

1.11 SUMMARY

This chapter includes a brief introduction to an overview of the project. And this chapter deals with the identifying problem statement, estimating the objective of the project, a brief introduction about the methodology used in the project, the significance of the project, organization of the project which includes the output of every chapter introduction. The next chapter is the Literature review which discusses various journal papers to obtain the specific problem statement by analyzing all the relevant work and information mentioned in that referencepaper to understand the present problem statement existing in that particular area.

CHAPTER 2

LITERATURE SURVEY

2.1 PREAMBLE

The purpose of the literature survey is to obtain a clear understanding of the existing problem in the particular area of the domain. Clearly understanding all the previousdevelopment and their works will provide the best way to obtain the perfect problem statement existing in the present condition. The following section summarizes the history of those works which were done previously, highlighting strengths and weaknesses of each method.

2.2 LITERATURE SURVEY

2.2.1 Based on Finger Print Biometrics

Zheng Hui Goh *et al.* [1] developed a system named as A Framework for Multimodal BiometricAuthentication Systems With Template Protection.The scheme that is used by this authoris Index-of-Max (IoM) hashing, Alignment-Free Hashing (AFH).By the study of this proposed system we can conclude that the performance of this scheme is Outlined a procedure that accepts both aligned and unaligned features of various origins and representations, and transforms them into a unified binarized cancellable template.And the data set is used in this proposed system is Fingerprint, 105 face, iris, and finger-vein.The Feature Conversion method used is cryptographic technique and gives security and privacy efficiency is high.The recognition Accuracy and Exhaustive Search is also high.Now going to Non-invertibility it does not supports ,whereas it supports the Revocability ,Cancellability and Unlinkability.

2.2.2 Based on Facial Image Biometrics

Hakyoung Lee *et al.* [2] suggested a system named as SoftmaxOut Transformation- PermutationNetwork for Facial Template Protection.The scheme that is used by this author is SoftmaxOut TransformationPermutation Network

(SOTPN). By this study of this proposed system we can conclude that the performance of this scheme is Introduced the permutable SoftmaxOut layer that integrates maxout units and a parameterized softmax function.And the dataset that is usedis LFW, YouTube Face and Face Scrub.The Feature Conversion method used is Arc face Networkand gives security and privacy efficiency is high. The recognition Accuracy and Exhaustive Searchis also high.Now going to features whereas it supports the Revocability , Cancellability .

2.2.3 Based on Multi-mode Biometrics

Jinjin Dong *et al.* [3] Processed the system named as Template Protection Based on DNA Coding For multimodal biometric recognition. The scheme that is used by this author is n DNA Coding For multimodal biometric recognition. By the study of this proposed system we can conclude that the performance of this scheme isThe proposed multimodal biometric template protection scheme does not affect the recognition performance and ensures the security of multimodal biometric templates. And the data set is used in this proposed system is Face ,Finger Prints veins.The Feature Conversion method used is DNA extractor and gives security and privacy efficiency is high.The recognition Accuracy and Exhaustive Search is also high. Now going to Non-invertibility ,Cancellability and Unlinkability it does not support ,whereas it supports the Revocability.

Loubna Ghammam *et al.* [4] Progressed the system named asEnhancing the Security of Transformation Based Biometric Template Protection Schemes.The scheme that is used by this author is BioHashing algorithm.By the study of this proposed system we can conclude that the performance of this scheme is This scheme can be used in real biometric authentication applicationsin industry as the computation is very fast while keeping a good privacy protection. And the data set used in this proposed system is Digital fingerprint and finger knuckle print images. The Feature Conversion method used is Feature Extractor and gives security and privacy efficiency is high.Therecognition Accuracy and Exhaustive Search is also high. Now going to Non-invertibility, Revocability, Cancellability it does not supports, whereas it supports the Unlinkability.

Simon Kirchgasser *et al.* [5] Elaborated the system named as Finger Vein Template ProtectionBased on Alignment-Robust Feature Description and Index-of-Maximum Hashing. The scheme that is used by this author is Index-of-Maximum Hashing. By the study of this proposed system wecan conclude that the performance of this scheme is Distance measures for the ARH feature vectorcould be investigated as well as the applied cosine measure might not be the bestperforming choice.And the data set is used in this proposed system is Finger Vein. The Feature Conversion method used is Feature Extractor and gives security and privacy efficiency is high. The recognition Accuracy and Exhaustive Search is also high. Now going to Non-invertibility ,Cancellability and Unlinkability supports ,whereas it does not supports the Revocability.

Yi C. Feng *et al.* [6] Fabricated the system named as A Hybrid Approach for Generating Secure and Discriminating Face Template. The scheme that is used by this author is THREE-STEP HYBRID .By the study of this proposed system we can conclude that the performance of this scheme is The proposed method not only protects the template but is also able to increase the template discriminability. And the data set is used in this proposed system is FERET, CMU-PIE, and FRGC.The Feature Conversion method used is cryptographic technique and gives security andprivacy efficiency is high.The recognition Accuracy and Exhaustive Search is also high.Nowgoing to ,whereas it supports the Non-invertibility . Revocability and does not support Cancellability, Unlinkability.

Sébastien Marcel *et al.* [7] proposed the Neural network-based face recognition with biometric template security. A face template is required to be known as a digital point of comparison for matching. The Deep Index of Maximum Hashing Method was the methodology employed for template protection. Any key or string of characters could be hashed to produce a different number. The initial string was typically represented with a shorter, fixed-length variableor key to make it simpler to locate or utilize. Implementing hash tables was the most popularapplication of hashing. The dataset used for implementation was made up of image dataof faces.Experimental findings on an unconstrained face dataset provide evidence for the efficacy of theproposed technique. The

fundamental flaw in this study was the assumption that stronger lossfunctions, which can reduce the intra-class variance of the hash code and broaden the tests to largerface datasets, are preferable.

Similarly, a deep index of maximum hashing for the face template protection scheme wasproposed by Jiandong cui *et al*. [8]. In this paper labeled faces data set was used for conducting experiments. The labeled faces were nothing but facial biometric authentications. A permutable deep feature extractor was applied to extract the features from the given dataset. Raw data were converted into binary data from the features using feature embedding techniques. Index hashing was used for security purposes. It was proved that this method provided high security and the efficiency was also high and the exhaustive search was moderate.

Arun Kumar Jindal *et al*.[9]. Developed A CNN-based approach for facial template protection was suggested by the authors. Convolutional neural networks (CNN/ConvNet) are a subclass often used in deep learning, deep neural networks evaluate visual input. The method proposed in this paper has enhanced the performance of matching. Face datasets namely CMU-PIE, FEI, and ColorFERET pictures make up the data sets used for experimenting. The development of template protection algorithms for behavioral biometrics including voice, keystroke dynamics, and large patterns has been expanded. A lot of security was offered. The rate of exhaustion was high.

A Jegede *et al* [10]. carried a novel face recognition and shielding functionality fortemplateprotection. The shielding function method was applied. The scheme suggested by the paper was togenerate a secret key that was difficult for a forger to access because neither it nor the biometric data were stored directly. A group of technologies that alter the source, bytes, or binary code of anapplication to fortify it against hacking, tampering, reverse engineering, and malware attacks. protected from manipulation and misuse that could cause a range of issues, such as unapproved entry, malicious code injections, password theft, app cloning, IP theft, larger-scale system threats, and more. The dataset for this study, which

consists of 200 face images, has a high level of securitybut a low level of efficiency. Although facial imagerecognition is highly effective, a thorough search is only marginally successful. The weaknesses of this work were a lack of resistance to spoofing attacks and sensitivity to record multiplicity attacks.

A new and novel privacy preserving of stable hash generation was developed by Dailé Osorio-Roig *et al*. [11] as a part of the template protection scheme. It used the face images as datasets and these face images were extracted from the DCNN method these extracted data get embedded by the face embedding system and here some homomorphic encryption was applied to the extracted data later onwards the hash generation generated a unique key for a particular data. These data were stored in the templates and these templates were stored in the databases. These systems was failed to give a high exhaustive search efficiency.

T.M. Dang *et al.* [12] suggested a scheme FEHash stands for Face Template Protection FullEntropy Hash. This study used a one-shot and multiple-shot enrollment strategy. The amount oftheactual quantity of information in the stream, which was measured in bits. If the hash was truly random, the entropy value will be the number of bits in the hash. The datasets used face photos from (CMU-PIE, FEI, and FERET). The benefit of this work is that, when measuring using the tunable matching method, it yields High (GARS) at the stringent operation at a point of zero FAR.The fundamental flaw in this paper is that a group of padding people was internally selected. However, the value of every bit can be ascertained if the attackers know all p individuals (such as their faces or embeddings). The security is very high and also the exhaustive rate is 100% genuine.The recognition rate was also high. High security wasprovided but the efficiency is low. The exhaustive search was very low.

To overcome the challenge of privacy-preserving preselection for protected biometric identification, Pia Bauspie *et al.* [13] proposed effectiveness in privacy-preserving biometric identification in the encrypted realm; By combining public-key encryption with keyword search. The authors employed the approaches of Enrolment, Identification, Privacy-Preserving Binning, Probe

Trapdoor Retrieval, and Discussion. For effective privacy-preserving biometric identification in the encrypted realm, the suggested approach combines PEKS and HE. The experiment on Privacy Preserving Preselection for Protected Biometric Identification was carried out by the authors, who used the data set execution times in seconds for a preselected subset of 1062 subjects' identification. They employed The De-Re-identification identified Risk Faces usedto evaluate the results of the de-identification process's protection performance The subset of the FERET face dataset with 963 subjects was used in the studies carried out by the authors. To guarantee that each participant had two color frontal face photographs (designated as"fa" and "fb"),this subset was selected from the images of count (994 subjects) that were available. It is noted that this study aims to finish the facial de-identification process by presenting a method for blending a de-identified face region with its original background. The proposed system blends these recognized facial regions with their original background while providing maximum privacy protection within the facial region.

Yi-Lun Pan *et al.* [14] accomplished another method of Multi-factor combination to enhance a reversible privacy protection system for face photographs. In comparison to existing methods, the suggested method can successfully anonymize face photos with high accuracy under the Anonymization and De-anonymization, Password Scheme, and Multi-factor Register Scheme in addition to a face image, a multi-factor attribute vector and a password as inputs. The following isa list of the experiments that the writers conduct, both statistically and qualitatively networks such as insightface-ir50 ms1m and face-ir50 Asia. In the experiment, the scheme created a ground- breaking, reversible privacy protection system that can automatically and graduallyanonymize and de-anonymize images using just one neural network. Without changing the data distribution, the proposed model can successfully de-anonymize photos using a range of parameters and anonymize images under different conditions. By employing a multi-factorsolution, the system was able to offer robust security protection.

A comprehensive framework for privacy protection for comprehensive

content-based information retrieval was suggested by Li Weng *et al. [15].* The authors were able to solve the issues by proposing the above method. It provided 2 levels of defense. Initially, reliable hashing values, and second, the client might decide to leave out specific bits from a hash value to give theserver even more ambiguity. Query Generation: Original content cannot be utilized in queries dueto privacy concerns. Database Indexing: The idea of piecewise inverted indexing serves as the foundation for database indexing. The authors evaluated the retrieval and privacy-preserving capabilities of a particular content recognition program. The discrete wavelet transform underliesthe other whereas arbitrary projections underpin the first. 50,000 facial images from the ImageNetpublic domain image of faces collection are housed in this repository (the validation set of ILSVRC2012). They each have 50 images and 1,000 categories. It should be highlighted that the authors propose the concept of "tunable privacy," which allows for policy-driven changes to the level of privacy protection. It is performed via hash-based piecewise inverted indexing.

Yi Wang *et al.* [16] framed a problem with a biometric database's similarity search that protects anonymity based on the essential resemblance in geometry The privacy-preserving similarity search problem is framed within the binary hypothesis testing paradigm. Inferences are the foundation of our privacy-preserving paradigm for similar searches in permitted and randomized Montgomery domains. An inference-based approach for privacy-preservingHammingspace similarity searches is described by the study's authors. The novel approach recommended hash-based indexing as another method for protecting data structures.

Luca Debiasi *et al.* [17] developed a model that Arnold changes happen anywhere from once and five times. To make the Arnold scramble transform easier, the image size was standardized. chooses a face from the ORL face database. The weaknesses of current face recognition systems are discussed, and the random forest and C4.5 decision tree algorithms are demonstrated for use in image recognition. The facial image scrambling capability of the SFR-RF model is strong.The following Dataset samples were used in the experiment by the

authors: FV-LED-Dorsal, (FV- Laser-Dorsal), and Thermal-Face at the top; HV-RL850-Palmar and HV-RL850-Dorsal at the bottom (from left to right). Data from the PROTECT Multimodal DB Dataset (PMMDB) were used in the tests. The PMMDB has information on the iris, face (visual light, NIR, 3D, and thermal), periocular, anthropometrics, and hand and finger veins of 69 individual participants. The suggestedimage's gray values are modified in a key-dependent manner using two CB schemes, also referredto as non-invertible many-to-one transformations.

A brand-new, effective face representation approach was developed by Wanli Xue *et al.* [18]. protects privacy in the Bloom filter space that satisfies resource constraints from IoT devices. Theproposed method maintained high data analytics while allowing analytics activities on privacy- preserving face data representation. A probabilistic data structure used to express set membership is the bloom filter. One Bloom filter may transmit and store one piece of data (a segment). If the requested data already existed in the Bloom filter, the query will result in the Bloom filter returningtrue (with a false positive). It was common practice to employ bloom filter-based encoding as an effective masking approach for string and categorical data computations. They choose SVD as theirfeature extraction technique for both face datasets. TheYale B data set was utilized to conduct experiments. The experimental results demonstrated that the suggested method can safeguard faceprivacy while simultaneously offering a high level of analytics value. For analytics jobs like similarity matching, regression, and classification, it would offer comparable utility.

Jian Wu Zhang *et al.*[19] suggested an Arnold transform used on face images, and the randomforest (RF) algorithm-based scrambled face recognition model is employed. The model used the RF classification algorithm to recognize scrambled faces by extracting features from doing so. Based on the random forest algorithm and face-scrambling picture attributes, the random forest scrambled face recognition (SFR RF) model. The model pulls features from the face database, generates label classification data, builds a training and testing set, and uses the RF classificationmethod to recognize scrambled faces. For comparison analysis,

Arnold transformations are done 1 to 5 times. To make the Arnold scramble transform easier, the imagesize was standardized. chooses a face from the ORL face database. The paper discusses the drawbacks of current face recognition systems and demonstrates the use of the C4.5 decision tree algorithm and random forest algorithm for image recognition. The facial image scrambling capability of the SFR-RF model is strong. The following Dataset samples used in the experiment by the authors: (FV-LED-Dorsal), (FV-Laser-Dorsal), and Thermal-Face are shown in the top row (from left to right), followed by (HV-RL850-Palmar)and (HV-RL850-Dorsal). The two CB methods that are suggested change the gray values of the image in a key-dependentway and are referred to as non-invertible many-to-one transforms.

A filter scheme that protects against the super-resolution, parrot, and inverse-filter attacks onprotected image regions was provided by Sarwar *et al.*[20]. The software is promoted as a tool that applies the specified privacy filter to videos and removes the jitter that the filter adds. The suggested approach provides roughly the same level of privacy protection as state-of-the-art privacy filters against naive assaults and the highest level of privacy protection against parrot, inverse-filter, and super-resolution attacks. The authors used cutting-edge facial recognition technology to evaluate GMM. They do not rely on an extra visual detector, such as stance, facialexpression, age, gender, or race, to defend against a parrot, an inverse filter, or a super-resolutionattack. AHCMM improved accuracy, but not as much as AGB, FGB, or even AGB, and it is moreimmune to inverse filter attacks than an SVGB even when utilizing a precise secret key. Even when the kernel size is relatively large, the SR attack can better reconstruct the faces. Particularly for AGB, FGB, and SVGB, parrot-IF attacks are more severeand significantly more accurate thannaive-IF attacks. AHGMM improves accuracy but less than (AGB, FGB, and (SVGB) and is moreresistant to an inverse filter attack even when using an accurate secret key.

Rohit Kumar Pandey *et al.*[21] developed a template security technique that uses common hashfunctions and MEB codes to handle the issue of uniformity and achieve template security.

The authors used deep CNNs to reduce the accuracy loss in template protection techniques.To address the problem of using faces as passwords, they choose for experimentation face databases gathered in regulated settings. They employ evaluation protocols such as modificationsin lighting, session, and attitude that would be typical of applications like face unlock because a decent level of user compliance is anticipated. The CMU PIE database includes 41,368 photos of 68 distinct individuals in 13 various stances, 43 various lighting scenarios, and 4 various expressions. In our research, we make use of 5 postures and all lighting nuances. The remaining photos are utilized for testing, and ten are picked at random for training. An information leak is prevented by the code's hash digest in the final protected template. The attacker database, which consisted of every frontal image from the Multi-PIE database, and the genuine database, which consisted of the smaller Yale database, were used to test the genuine and impostor distributions using a dictionary attack. We achieved high (95%) GARs at the tight operational point of zero FAR, proving that the superior performance of deep CNNs can be used to lessen the loss of matching precision caused by template protection techniques.

Jing Jing Yang et al. [22] carried out a way to keep user privacy protected while maintaining access to their facial photos. The method performed better than other approaches of a similar naturein terms of quality, running speed, and target identification precision. The authors employed convolutional layers in their work. The high-accuracy models have already been properly trained on the datasets VGG FACE and VGGFACE2. The facial recognition model that had been correctly trained was referred to as the target face recognition network. The difference between the trainingset, development set, and the test is 98:1. The VGG FACE datasets contain 2622 people with various identities. Improving model stability and accelerating convergence speed. Both AdvGAN and PcadvGAN can generate facial images that ensure "availability. ’Furthermore, when comparedto AdvGAN and PriGAN, Pcadv GUN produces higher-quality facial images. The availability of their facial photos while protecting their privacy. The solution exceeds competing methods in termsof target recognition network accuracy, processing

speed, and image quality. When users share images on social media, Lihong Tang *et al.*[23]suggested an automatic tagging system to preservetheir anonymity. To enhance the functionality of the broad searching module, include additional information retrieval techniques. When users share images on social media, Lihong Tang *et al.*[23]suggested an automatic tagging system to preserve their anonymity. To enhance the functionality of the broad searching module, include additional information retrieval techniques.

The authors used face identity initialization in their work. Inferring from the survey data, It is found that:

Table.2.1: Feature Analysis Table

Authors	Dataset	Feature Conversion Method	Method	Security	Privacy Efficiency	Recognition Accuracy	Exhaustive Search
ENG HUI GOH[1]	Fingerprint, 105face,iris, and finger-vein	cryptographic technique	Index-of-Max (IoM)hashing, Alignment-Free Hashing (AFH)	Yes	100%	100%	98.9%
HakyoungLee [2]	LFW, YouTube Face and Face Scrub	Arc faceNetwork	SoftmaxOut Transformatio nPermutation Network (SOTPN).	yes	99.8%	100%	90%
Jinjin Dong[3]	Face,FingerPrints veins	DNA extractor	DNA CodingFor multimodal biometric recognition	yes	86.6%	98.9%	82.2%
Loubna Ghammam[4]	Digital fingerprint andfingerknuckle print images	Feature Extractor	BioHashing algorithm	yes	57.4%	96.0%	59.9%
Simon Kirchgasser[5]	Finger Vein	Feature Extractor	Index-of- Maxi mumHashin g	Yes	99.9%	100%	99.8%
Yi C.Feng[6]	FERET, CMU-PIE, andFRGC	Crypto Encryption	THREE-STEP HYBRID	Yes	100%	98.0%	100%

2.3 GAP IDENTIFICATION

In previous methodology hashing is not used highly but we can say that by using using hashingmethod the security and privacy must be increased highly The most challenging problem is to protectthe plain biometrics received during the enrolment time. Some papers explained thatproposed systemneeds huge storage requirements to store the features of faces and other biometrics. In some papers

there is vulnerable recognition of faces and biometrics. In some methods proposed method have to follow potential privacy issues to store the data. When an attacker is aware of the biometric symmetry of two or more biometric templates used in a multi-biometrics.

CHAPTER 3

SYSTEM REQUIREMENT

3.1 PREAMBLE

The previous chapter describes an examination of the research on template protection schemes for secure the multimodal biometrics using hash generation under different template protection schemes , together with the benefits and drawbacks of those approaches. This chapter describes the various kinds of tools used to create the system proposed for pre requisites. This taskdiscusses the tools used to design the proposed system, such as front-end tools, back-end tools, hardware tools, and various network requirements.

3.2 FRONT END TOOLSPyCharm

PyCharm provides smart code completion, code inspections, on-the-fly error highlighting PyCharm offers great framework-specific support for modern web development frameworks such as Django, Flask, Google App Engine, Pyramid, and web2py.ng and quick fixes, along with automated code refactoring and rich navigation capabilities. PyCharm integrates with Python Notebook, has an interactive Python console, and supports Anaconda as well as multiple scientific packages including matplotlib and NumPy.

3.3 COMMUNICATION INTERFACE

Python is a general-purpose interpreted, interactive, object-orientated, and high-level programming language. It changed into created with the aid of using Guido van Rossum at some point in 1985- 1990. Like Perl, Python supply code is likewise to be had below the GNUGeneral Public License (GPL). Python is a high-level, interpreted, interactive, and object orientated scripting language. Python is designed to be tremendously readable. It makes use ofEnglish key phrases often

while different languagesuse punctuation. Some of the features of the python are discussed below.

Python is Interpreted - Python is processed at runtime with the aid of using the interpreter. You do now no longer want to assemble your software earlier than executing it. This is much like PERL and PHP.

Python is Interactive - You can take a seat down at a Python activateand have interaction with the interpreter immediately to jot down your programs.

Python is Object-Oriented − Python supports the Object-Oriented style or technique of programming that encapsulates code within objects.

Python is a Beginner's Language − Python is a great language for beginner level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

3.4 OPERATING SYSTEM

It is an interface between a computer user and computer hardware. An operating system issoftware that performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drivesand printers. It is a time-sharing operating system that schedules tasks for efficient use of the system and may also includeaccounting software for cost allocation of processor time. In this proposed work we useWindows operating system. Windows is a series of operating systems developed by Microsoft. Each version of Windows includes a graphical user interface, with a desktopthat allows users to view files and folders in windows. For the past two decades, Windows has been the most widely used operating system for personal computers and PCs. We can use the versions such as Windows 7 and the above versions such as Windows 10.

3.5 HARDWARE REQUIREMENTS

3.5.1 Processor

A processor is an integrated electronic circuit that performs the calculationsthat runthe computer. A processor performs arithmetical, logical, input/output (I/O), and other basic instructions that are passed from an operating system (OS). The

Pentium Dual-Core brand was used for mainstreamx86 architecture microprocessor from Intel from 2006 to 2009 when it was renamed Pentium. The processors are based on either the 32-bit or 64-bit, and it was targeted at mobile or desktop computers. In terms of features, price, and performance at a given clock frequency, Pentium Dual-Core processors were positioned above Celeron but Core and Core2 microprocessor in Intel's product range. The Pentium Dual-Core was also a very popular choice for overclocking, as it can deliver high performance (whenoverclocked) at a low price.

3.5.2 RAM (Random Access Memory)

Random-access memory (RAM) is a form of computer memory that can be read and changed in any order, typically used to store working data and machine code. A RAM device allows data items to be read or written in almost the same amount of time irrespectiveof the physical location of data inside the memory. RAM contains multiplexing and demultiplexing circuitry, to connect the data lines to the addressed storage for reading or writing the entry. Usually, more than one bit of storage is accessed by the same address.

3.5.3 HARDWARE REQUIREMENTS TOOLS

- System: Intel or compatible Pentium dual-core
- Hard Disk: More Than 500 GB
- Memory (RAM): At least 8GB
- OS: Windows 7 or 7+
- System type: 64-bit Operating System

3.5.4 SOFTWARE REQUIREMENTS

- Coding Language: Python 3.6 or high version
- IDE: PyCharm
- For Audio: pygame python module is used

Python Libraries

NumPy: NumPy is the fundamental package for scientific computing inpython. NumPy arrays give advanced mathematical and other types of operations on large numbers of data. Typically, such operations are executed sequences.

Matplotlib: Matplotlib. Py plot is a collection of functions that make matplotlib work like MATLAB. Each py plot function makes some change to a figure: e.g., creates a figure, createsa plotting area in a figure, plots some lines in a plotting area, decorates the plotwith labels, etc.

Pandas: Pandas are mainly used for data analysis. Pandas allow all importing data from variousfile formats such as comma-separated-values, JSON, SQL,and Microsoft Excel. Pandas allow various applications that are data manipulation operations such as merging, reshaping, selecting, as well as data cleaning, and data wrangling features.

Scikit-Learn: Scikit-learn is a free machine learning library for the Python programming language. It features various algorithms like support vector machines, random forests, and k- neighbors, and it also supports Pythonnumerical and scientific libraries like NumPy .

Tensorflow: Tensor Flow is an open-source library for fast numerical computing. It was created and is maintained by Google and released under the Apache 2.0 opensource license. The API is nominally for the Python programming language, although there is access to the underlying C++ API.TensorFlow was designed for use both in research and development andin production systems.

Open CV: Open CV-Python makes use of NumPy, which is a highly optimized library fornumerical operations with a MATLAB-style syntax.

3.6 SUMMARY

This chapter introduced the software and hardware requirements of the system. The above requirements are required for the successful implementation of the project. Python is theprogramming language chosen for the implementation of PE detection, for better implementation and accurate results using most PyCharm. A system with the 64-bitWindows operating system and an Intel core Processor device. The hardware requirementsare the basic ones that are required for the execution of any regular python script.

CHAPTER-4

SYSTEM DESIGN

4.1 PREAMBLE

The limitation was that the experiments for several image quality ranges showed thatfaceimage quality has a significant impact on the proposed system. However, the experimentalevaluation over the LFW showed that for high quality images not for low quality images. APfinds its best examples on a latent space of face embeddings (by using distance metrics)

4.2 ARCHITECTURE DIAGRAM

This architecture was fundamental for any template protection scheme. The first stepwas taking the dataset as images of human faces for any biometrics authentications. Then these images were selected for training, testing, and assessing the effectiveness of machine learning and artificial intelligence (AI) algorithms, also known as common vision algorithms, which areincluded in an imagedataset.

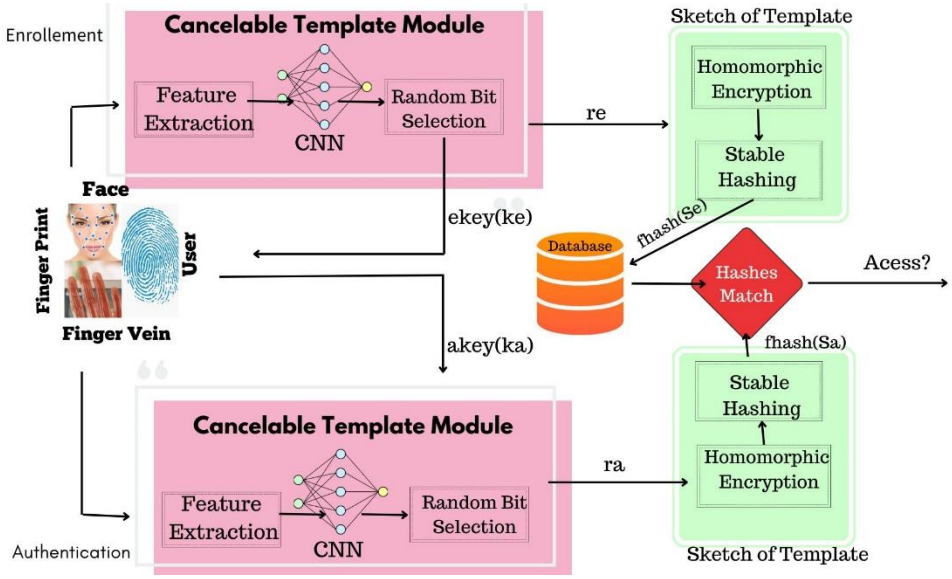


Fig 4.1 Architecture Diagram

Features were extracted by some feature-extracting methods like CNN, Vector, etc. Now the features get Embedded were converted into unintelligible format, which means a method of encryption.By using a secret key, picture encryption transforms a plain image into an encrypted one. CNN is usedas the following step in the data acquisition process. Layers ofa deep convolutional neural network. Convolutional and pooling layers are usually switched back and forth. Convolutional neural networks with deep learning.

The above architecture was given a clear idea about how the templates were protectedand how the data get enrolled during the enrollment phase and the newly generated templateswere verified during the authentication phase.The enrollment and authentication phases were explained in the following section.

4.2.1 Enrollment

During the enrollment phase, face biometrics were captured for the users who want to enroll forany authenticationservice and to become legitimate users for that system. The face image dataset isgiven to the feature extractor to extract the features of the faces by using any feature extraction algorithm like CNN. This neural network induces thefeatures of the face and that features are goingto be encrypted. This Encryption technique was used to convert the features into a code language (Binary data) nothing but raw data into encrypted binary data. The further step is Hash key generation. This hash key generation is used to generate a unique key for a particular data for providing security.This key generated data gets stored in the Templates. Now the final step in this enrollment stage is the Templated is stored in the database. Hash keys were generated from the features randomly. Because if the key was compromisedby the attacker, then it is difficult to changethe face biometric unlike the password, the face might not be changed.

4.2.2 Authentication

During the authentication phase, the face image of the legitimate user was given. The features were extracted from the face image of the querying user. The same encryption technique was usedto encrypt the features into a codelanguage (Binary data) nothing but raw data into encrypted binarydata. Then the biometric

key is generated from thefeatures randomly. From that hash, keys were generated. This hash key is used to generate a unique key for a particulardata for providing security.This key generated data gets stored in the Templates. Now the last step here compares thehash key generated during the enrollment and the hash key generated during the authentication stage. The original data template which means the enrollment stage and the retrieval stage template means theauthentication stage gets matched. If the matching rate is higher, then the user is authorized to accessdata otherwise prevented from access.

4.3 Feature extraction:

Feature extraction is a crucial step in biometric systems that involves identifying and extracting unique characteristics or features from biometric data such as facial images, fingerprints, and finger veins. Hereare some common techniques used for feature extraction in each of these biometric modalities:

Facial Image:

- Local Binary Pattern (LBP) - A texture-based feature extraction method that extracts textureinformationfrom local image regions.
- Scale-Invariant Feature Transform (SIFT) - A feature extraction method that identifies scale- invariant keypoints and extracts descriptors around them.
- Histogram of Oriented Gradients (HOG) - A method that captures information about the shapeanddistribution of edges in an image.

Fingerprint:

- Minutiae-based features - The most common type of fingerprint feature extraction technique, whichidentifies and extracts characteristics such as ridge endings, bifurcations, and ridge paths.
- Singular Points - Points on the fingerprint image where the ridges either terminate or bifurcate andcan beused as landmarks.
- Texture-based features - Extracting texture information from the image using methods such asGaborfilters or Local Binary Pattern (LBP).

Finger Vein:

- Local binary pattern (LBP) - This method is also used for finger vein

feature extraction, where itextractsthe texture features of vein patterns.

- Discrete Wavelet Transform (DWT) - A method that decomposes the vein image into differentfrequencysub bands, allowing for the extraction of vein patterns at different scales.
- Gabor filter - Another method that can extract texture features by convolving the vein image with a set of Gabor filters that capture different frequency and orientation information.

Overall, the goal of feature extraction is to capture the unique characteristics of each biometric modality in a way that is robust, discriminative, and efficient.

4.4 Homomorphic Encryption

Homomorphic encryption is a cryptographic technique that allows for computations to beperformed on encrypted data without first having to decrypt it. This makes it an ideal solution forprotecting sensitive biometric data while still allowing for computations to be performed on it. Inthe context of multimodal biometrics, homomorphic encryption can be used to protect data from different modalities such as facial images, fingerprints, and finger veins. Here are some ways homomorphic encryption can be used for multimodal biometrics:

Secure biometric data sharing: Homomorphic encryption can be used to securely share biometric data across different organizations or systems. For example, a hospital could encryptand share a patient's biometric data with a specialist or another healthcare provider without revealing the underlying biometric data.

Multi-party biometric authentication: Homomorphic encryption can be used to enable multi- party biometric authentication, where multiple parties (e.g., a bank and a mobile network operator) need to verify a user's identity before granting access to a service or account.

Secure biometric matching: Homomorphic encryption can be used to perform biometricmatching on encrypted data, without first having to decrypt the data. This can help protect sensitivebiometric data from being exposed in the event of a security breach or data leak.

Secure biometric data storage: Homomorphic encryption can be used to protect biometricdatastored in the cloud or on a remote server, making it more difficult for attackers to gain accessto sensitive data.

Overall, homomorphic encryption is a promising technology for enhancing the security and privacy of multimodal biometric systems, while still allowing for computations to be performedonsensitive biometric data.7

4.5 Hash Key Generation

Hash key generation is an important step in the process of securing biometric data inmultimodal biometric systems. A hash key is a fixed-length string of bits that is generated from the biometric data and used to securely store and authenticate the data. Here are some techniques that can be used to generate hash keys for multimodal biometric data:

Feature-based hashing: This method involves first extracting features from thebiometric data and then using these features to generate a hash key. For example, in the case of facial images, features such as eyespacing, nose shape, and mouth shape could be extractedand used to generate a hash key.

Biometric template-based hashing: This method involves generating a biometric template from the raw biometric data, and then using the template to generate a hash key. The template contains a reduced representation of the biometric data that can be used to compare and authenticate the data.

Multimodal biometric hashing: This method involves combining data from multiple biometric modalities to generate a hash key. For example, in a multimodal biometric system that uses both fingerprint and face recognition, the data from both modalities could be combined to generate a more secure hash key.

Randomized hashing: This method involves generating a random string of bits and then combining this string with the biometric data to generate a hash key. This technique can be used to enhance the security ofthe hash key by introducing an element of randomness.

Overall, the goal of hash key generation is to create a fixed-length, secure

representation ofthe biometric data that can be used for authentication and storage. The specific technique used for hash key generation will depend on the characteristics of the biometric data and the requirements of the multimodal biometric system.

4.6 Template Protection

Template protection refers to the process of securing biometric templates used in biometricsystems against attacks and unauthorized access. A biometric template is a mathematicalrepresentation ofa biometric trait, such as a fingerprint or facial image, that is stored in a database and used for subsequent biometric matching.

Decoding refers to the process of translating or converting encoded data back into its original, human-readable form. Encoded data is typically in a format that is not directly understandable by humans, but can be easily processed by computers or other devices. Decoding is necessary to retrieve the original data and make it usable for human consumption.

4.6.1 Cryptographic Hashing

The hash function is designed to be a one-way function, meaning that it is easy tocompute the hash value from the input data, but it is computationally infeasible to compute the original input data from the hash value. This property makes cryptographic hashing useful for storing passwords, verifying the integrity of data, and digital signatures. Here are some properties of cryptographic hashing:

Deterministic: Given the same input data, a cryptographic hash function will alwaysproduce the same hash value.

Uniqueness: Even a small change in the input data should produce a significantly different hash value.

Non-reversibility: It should be computationally infeasible to reverse the hash function to recover theoriginal input data.

Collision resistance: It should be computationally infeasible to find two different input data that producethe same hash value.

Commonly used cryptographic hash functions include SHA-256, SHA-512, and MD5. These hash functionsare widely used for securing passwords.

4.7 Applications

- Law enforcement and border control
- Healthcare
- Natural language processing
- Banking and financial services
- Physical access control Consumer electronics

4.8 SYSTEM REQUIREMENTS

Requirement's analysis is very critical process that enables the success of a system or software project to be assessed. Requirements are generally split into two types: Functional and non-functionalrequirements.

4.8.1 Functional Requirements

These are the requirements that the end user specifically demands as basic facilities thatthe system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be givento the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

Examples of functional requirements:

- ➢ Authentication of user whenever he/she logs into the system.
- ➢ System shutdown in case of a cyber-attack.
- ➢ A verification email is sent to user whenever he/she register for the first time on some software system.

4.8.2 Non-functional Requirements

These are basically the quality constraints that the system must satisfy according to the projectcontract. The priority or extent to which these factors are implemented varies from one project to other.They are also called non-behavioral requirements.

They basically deal with issues like –

- Portability
- Security
- Maintainability
- Reliability

- Scalability
- Performance
- Reusability
- Flexibility

4.9 SYSTEM DESIGN

Input Design is the one in which in the input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc. Therefore, the quality of system input determines the quality of system output.Well-designed input formsand screens have following properties

- It should serve specific purpose effectively such as storing, recording, andretrieving the information.
- It ensures proper completion with accuracy.
- It should be easy to fill and straightforward.
- It should focus on user's attention, consistency, and simplicity.

4.9.1 Objectives for Input Design

- To develop output design that serves the intended purposeand eliminates the production of unwanted output.
- To develop the output design that meets the end user's requirements.
- To deliver the appropriate quantity of output.
- To form the output in appropriate format and direct it to the right person.
- To make the output available on time for making good decisions

4.9.2 Objectives of Output Design

- To develop output design that serves intended purpose andeliminatesproduction of unwanted output.
- To develop the output design that meets the end user's requirements.
- To deliver the appropriate quantity of output.
- To form the output in appropriate format and direct it to the rightperson.
- To make the output available on time for making good decisions.

4.10 SUMMARY

This chapter deals with the system design. System architecture, data flow diagrams are all included. System architecture is a conceptual framework that describes the organization, behavior, and system parts that will cooperate to construct the whole system. System designing is very important in order toimplement the model.

CHAPTER 5

METHODOLOGY

5.1 PREAMBLE

The previous chapter describes the design of the proposed system related to the Template Protection Scheme for Secure the Multimodal Biometrics Using Hash Generation. The next step after designing is the implementation of the proposed approach. This chapter deals with the implementation of the approach which is proposed. The current section consists of the proposed approach and various steps required implementing the new system. It gives a brief view of the implementation part.

5.2 Template Protection Scheme for Secure the Multimodal Biometrics Using Hash Generation

The goal of template protection schemes is to ensure that the biometric templates cannot be reconstructed from the protected data, even if the protected data is compromised. Templateprotection schemes typically involve the use of cryptographic techniques such as hashing, encryption, and key generation.

In a template protection scheme, the biometric template is first transformed into a fixed-length bit string using a secure cryptographic hash function. The hashed template is then encrypted using a key that is unique to each user, and the encrypted template is stored in a secure database or on a smart card.

When a user wants to authenticate or identify themselves, their biometric data is captured and transformed into a biometric template, which is then compared to the stored template using a comparison algorithm. The comparison algorithm is designed to compare the encrypted and hashedtemplates, rather than the raw biometric data. If the comparison algorithm indicates a match, the user is authenticated or identified.

Template protection schemes provide several benefits over traditional biometric authentication methods. They help to protect the privacy of users by ensuring that their biometric data cannot bereconstructed from the protected data. They also make it more difficult for attackers to perform replay attacks or brute-force attacks, as they would need access to the user's key to decrypt the hashed template. Overall, template protection schemes provide a secure and effective way to use biometric data for authentication and identification purposes.

5.3 EXPLAINATION OF TEMPLATE GENERATION AND PROTECTION

Designing a Template Protection Scheme (TPS) for securing multimodal biometrics using hashgeneration involves the following steps:

Biometric Data Acquisition: The first step is to acquire the biometric data, which can be done using various sensors such as fingerprint scanners, iris scanners, facial recognition cameras, etc.

5.3.1 Feature Extraction:

The next step is to extract features from the acquired biometric data. This involves identifying and isolating specific characteristics that are unique to the individual and can be usedfor biometric authentication.

5.3.2 Hash Generation:

The extracted features are then hashed using a one-way function that generates a uniquehash valuefor each individual's biometric data. This hash value should be sufficiently long andrandom to prevent reverse-engineering and collision attacks**.**

5.3.3 Template Generation:

The hash values are then used to generate templates, which are stored in a secure database. These templates act as a representation of the individual's biometric data and are used for authentication.

5.3.4 Authentication:

During authentication, the user's biometric data is again hashed and compared with the stored template's hash value. If the hash values match, the user is authenticated, and access is granted.

5.3.5 Revocation:

In case of compromise of the template, a new template is generated for the user,and the compromised template is revoked.

5.3.6 Security Analysis:

Finally, the TPS must undergo thorough security analysis to ensure it is resistant to attacks such as brute force, collision attacks, and reverse engineering.In conclusion, designing a TPS for securing multimodal biometrics using hash generation involves acquiring biometric data, extracting features, generating hashes, creating templates, authenticating users, revoking compromised templates, and performing a security analysis.

5.4 DATA SET USED

When developing a multimodal biometric system that utilizes face, fingerprint, and finger veindata, there are several datasets that can be used for testing and evaluation. Here are a few commonly used datasets:

Multi-Modal Face Database (MMFD): This database contains facial images of 100 subjectscaptured under three different lighting conditions. It also includes fingerprint and finger veinimages of the same subjects.

FVC-on Going: This is an ongoing fingerprint verification competition that provides a standardized dataset for testing and comparing fingerprint recognition algorithms.

Poly U FV-GMS database: This database contains finger vein images from 280 subjects, capturedusing a near-infrared camera. The database includes both left and right index fingers for each subject.

CASIA FaceV5: This dataset contains facial images of 1,040 subjects, captured under differentlighting conditions, facial expressions, and poses. It also includes 2Dand 3D facial images, as well as facial thermograms.

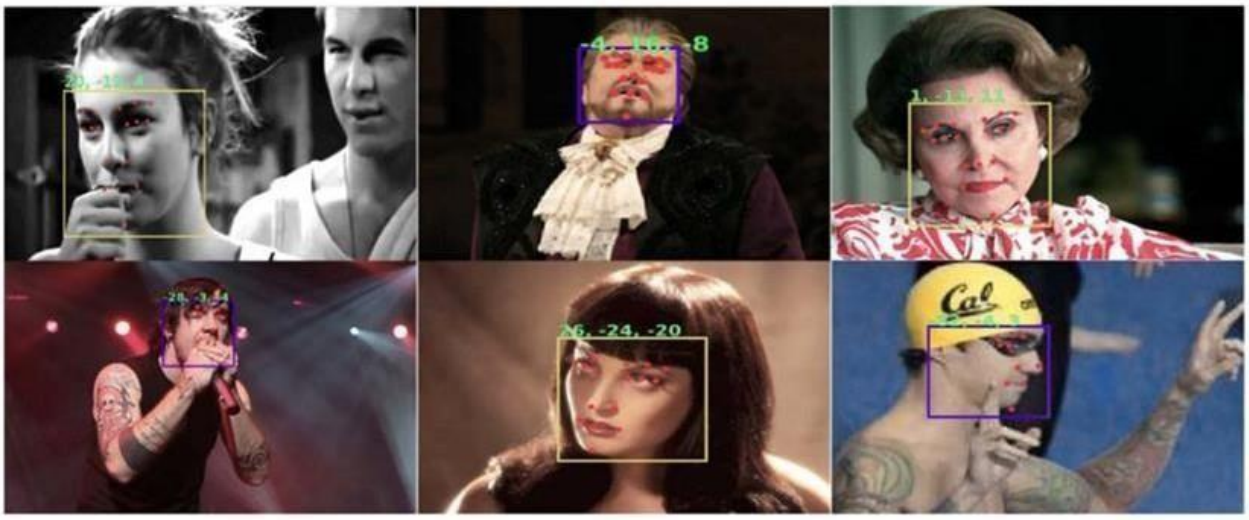
Bio secure ID: This dataset includes both face and fingerprint data from 500 subjects. The dataset includes both indoor and outdoor environments, and the images are captured using different cameras and sensors. There are also several other publicly available datasets that can be used for developing and testing multimodal biometric systems. It is important to choose a dataset that is appropriate for the specific application and use case, and that provides a realistic representation ofthe types of biometric data that will be used in the system.

5.4.1 Facial Dataset

Facial recognition is one of the most widely used biometric modalities, and thereare many datasets available for training and evaluating facial recognition algorithms. Here are some commonly used facial datasets:

Labeled Faces in the Wild (LFW): This dataset contains more than 13,000 facial images of 5,700 subjects, captured under natural conditions. The dataset is widely used for testing and comparing facial recognition algorithms.

Celeb A: This dataset contains more than 200,000 facial images of celebrities, along withannotations such as facial landmarks and attributes.



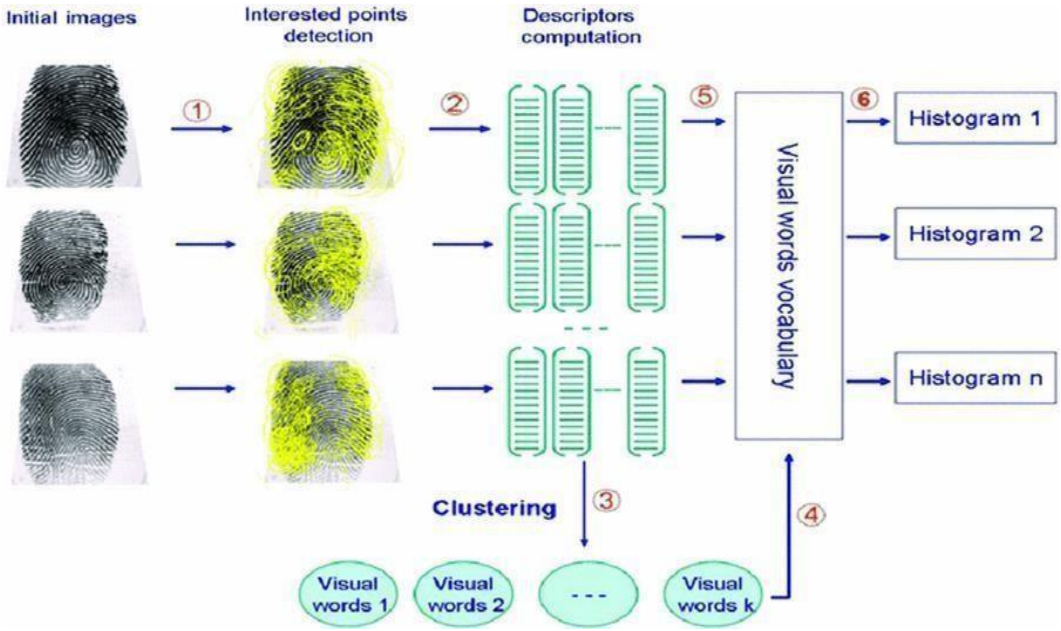
There are also many other facial datasets available, ranging from small-scale datasets designedfor research purposes to large-scale datasets designed for commercial applications. When choosing a facial dataset, it is important to consider factors such as the size of the dataset, the quality and diversity of the images, and the availability of annotations and metadata.

5.4.2 Finger Print Dataset

Fingerprint recognition is another widely used biometric modality, and there are many datasets available for training and evaluating fingerprint recognition algorithms. Here are some commonly used fingerprint datasets:

Fingerprint Verification Competition (FVC): This is a series of fingerprint verification competitions that provide standardized datasets for testing and comparing fingerprint recognition algorithms. The most recent competition is FVC-on Going, which includes datasets collected under different conditions and with different sensors.

Fingerprint Image Segmentation and Quality Estimation (FISQ) Database: This database contains fingerprint images collected using a range of sensors and imaging conditions. The database includes ground-truth segmentations and quality scores, making it useful for testingsegmentation and quality estimation algorithms.



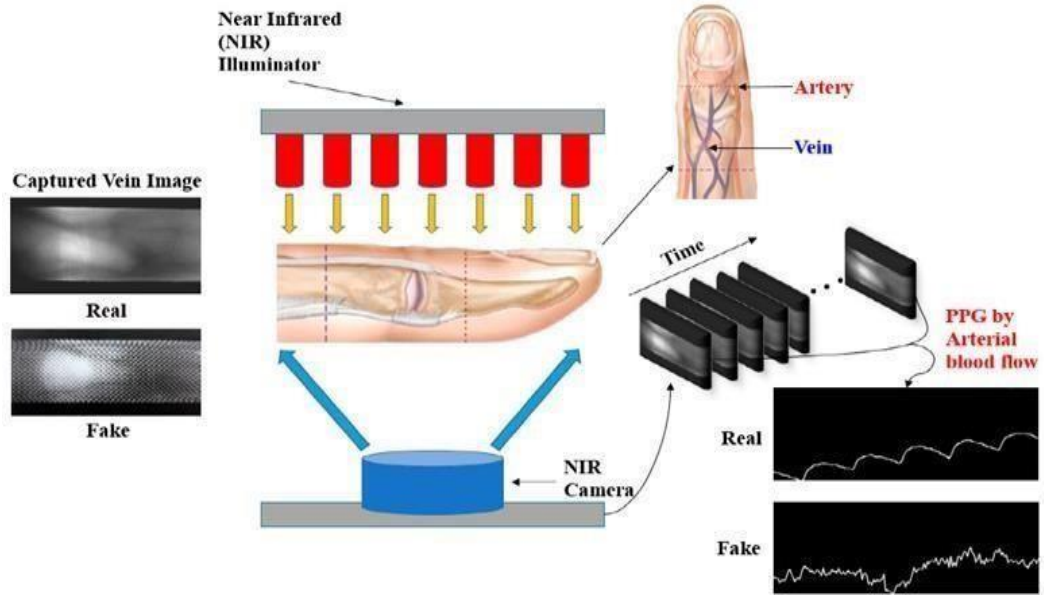
When choosing a fingerprint dataset, it is important to consider factors such as the size of the dataset, the quality and diversity of the images, and the availability of annotations and metadata.

5.4.3 Finger Vein Dataset

Finger vein recognition is a relatively new biometric modality, and there are fewer publicly available datasets for training and evaluating finger vein recognition algorithms. Here are somecommonly used finger vein datasets:

Poly U Finger Vein Database (Poly U-FV): This database contains finger vein images from 280 subjects, captured using a near-infrared camera. The database includes both left and rightindex fingers for each subject, and it is widely used for testing and comparing finger vein recognition algorithms.

Finger Vein Verification Competition (FVVC): This is a series of finger vein verification competitions that provide standardized datasets for testing and comparing finger vein recognition algorithms. The most recent competition is FVVC2018, which includes datasetscollected using different sensors and imaging conditions.



When choosing a finger vein dataset, it is important to consider factors such as the size of the dataset, the quality and diversity of the images, and the availability of annotations and metadata.It is also important to consider whether the dataset includes images captured under different conditions and with different sensors, as this can help to evaluate the robustness and generalizability of finger vein recognition algorithms.

5.5 IMPLEMENTATION

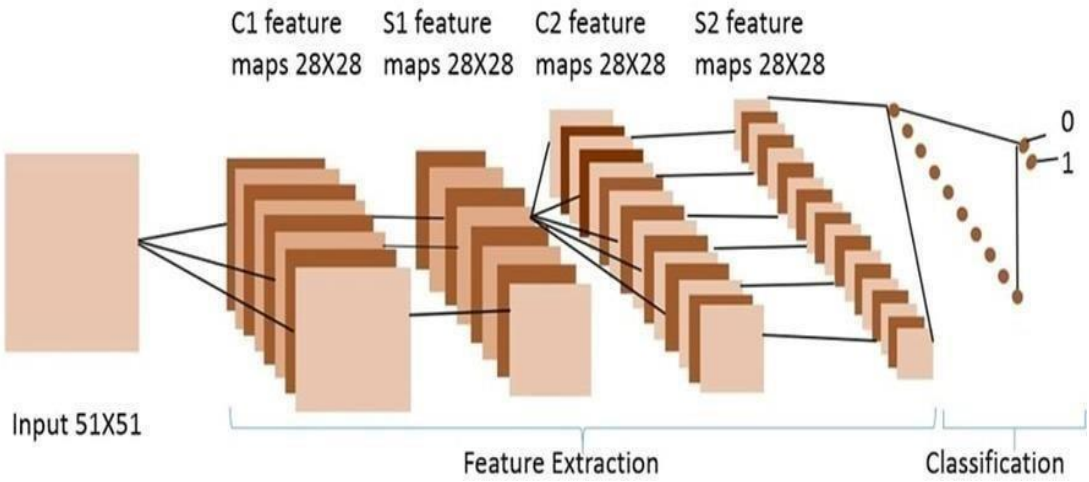
Implementing template protection schemes involves a series of steps, including feature extraction, template generation, hash generation, and encryption. The relevant biometric featuresare extracted and transformed into a standardized template representation, which is then hashed to generate a fixed-length digital fingerprint.

The hash value is encrypted and stored securely in a database, while the original biometric data is discarded to protect the privacy of the individuals. During recognition, the input biometric datais processed to generate a new template, which is hashed and compared against the encrypted templates in the database to find a match.

Template protection schemes ensure the privacy and security of the biometric data and preventthe risk of identity theft and unauthorized access. The implementation of template protection schemes requires careful consideration of various factors, such as the choice of hash function,encryption algorithm, and key management.

5.5.1 Feature extraction

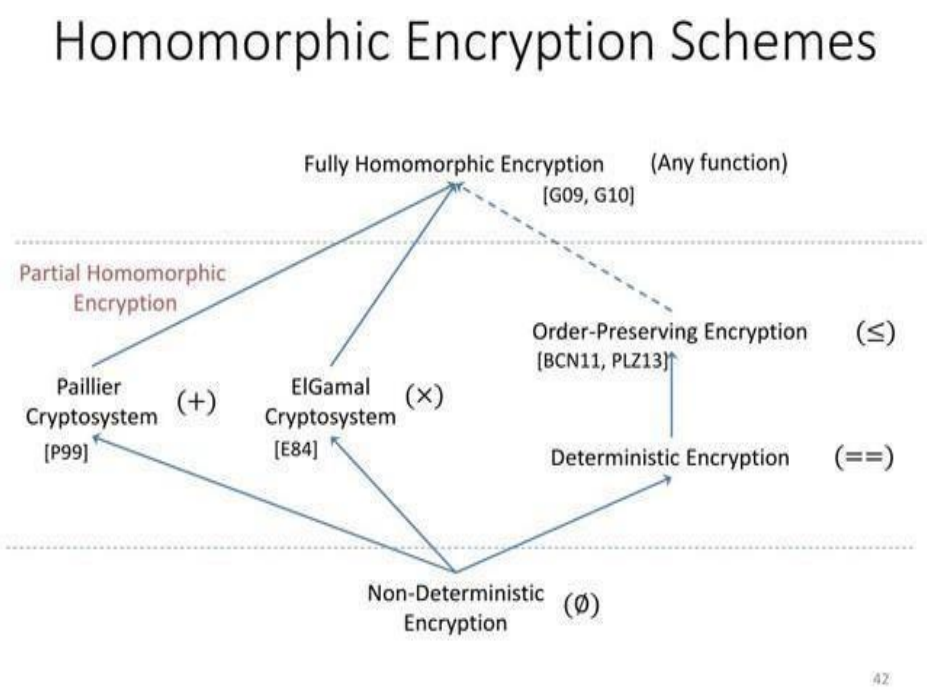
Feature extraction in multimodal biometric recognition involves extracting relevant and discriminative features from the different biometric modalities. This is typically done using techniques such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA),or Convolutional Neural Networks (CNNs).



The extracted features are then combined into a single feature vector or template, which is usedfor recognition. The goal of feature extraction is to reduce the dimensionality of the data while preserving the most relevant information, and to ensure that the resulting features are robust anddiscriminative across different modalities and conditions.

5.5.2 Homomorphic Encryption

Homomorphic encryption is a type of encryption that allows computation to be performed on ciphertext, without first decrypting it. This means that data can be encrypted and stored on remote servers, and computations can be performed on the encrypted data without ever revealingthe plaintext. Homomorphic encryption has applications in secure cloud computing, privacy- preserving data mining, and secure machine learning, among others.

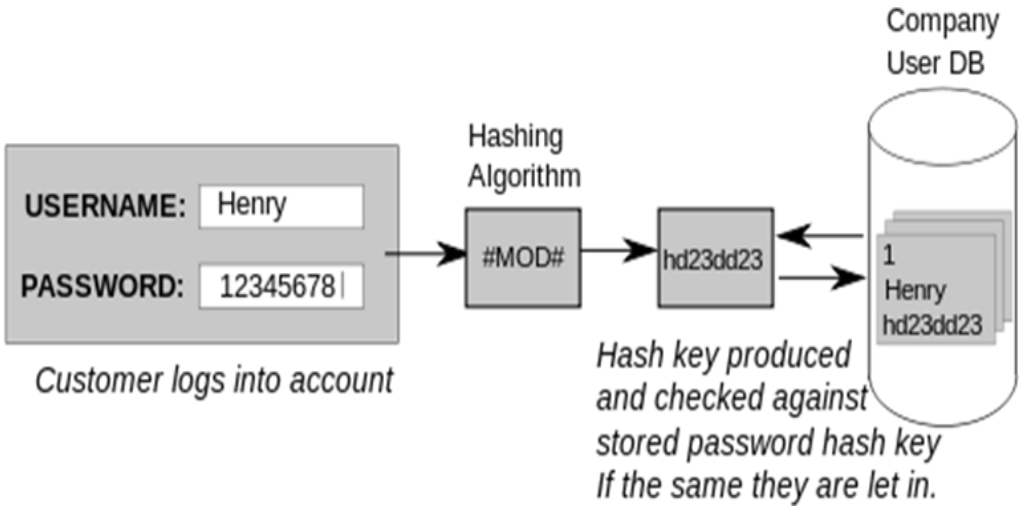


There are different types of homomorphic encryption schemes, such as fully homomorphicencryption (FHE), partially homomorphic encryption (PHE), and somewhat homomorphic encryption (SHE), each with different trade-offs between security, efficiency, and functionality.

While homomorphic encryption is still an active area of research, recent advances in algorithmsand hardware have made it more practical for real-world applications.

5.5.3 Hash key Generation

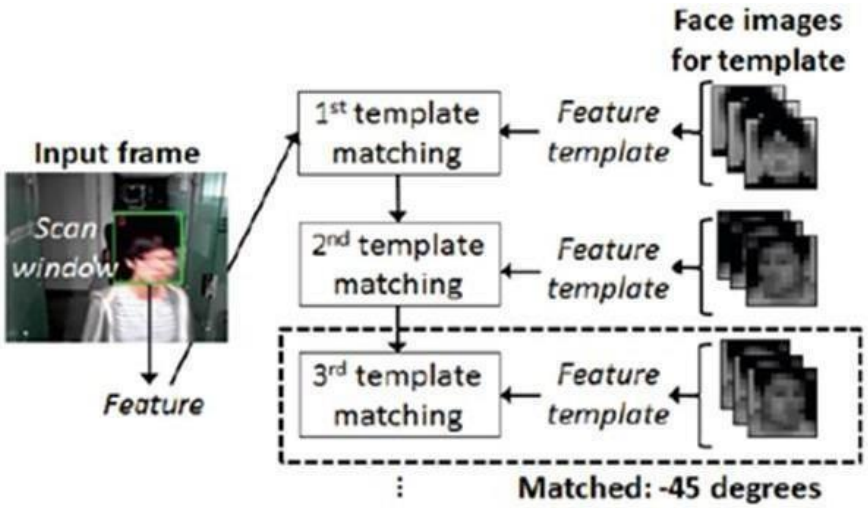
Hash key generation is the process of generating a unique, fixed-length digital fingerprint or summary of a piece of data, such as a file or message. This is typically achieved using cryptographic hash functions, which are mathematical algorithms that take input data of any sizeand produce a fixed-length output (known as the hash value or message digest).



The hash value is unique to the input data and is used to verify the integrity and authenticity of the data. Hash key generation is widely used in various applications, including data encryption, digital signatures, password storage, and message authentication.

5.5.4 Template Generation

Template generation is the process of extracting and representing biometric features from raw biometric data to create a compact and standardized representation of an individual's biometric traits. The extracted features are often transformed and normalized to improve their quality and reduce the impact of environmental and sensor variations.



The resulting template is a compact and secure representation of the biometric data that can be stored and compared for biometric recognition purposes. Template generation is a crucial step inbiometric recognition systems, as it directly affects the accuracy, security, and privacy of the system. Different biometric modalities may require different techniques for template generation,but the goal is always to create a robust and discriminative representation of the biometric traits that can withstand variations in acquisition conditions and be used for accurate and efficient recognition.

5.5.5 Database

Multimodal biometric systems typically store the biometric data of individuals in a database in the form of templates or feature vectors. These templates are generated by extracting relevant anddiscriminative features from the different biometric modalities, such as face, fingerprint, andiris. The templates are often encrypted and securely stored in the database to prevent unauthorized access and protect the privacy of the individuals.

During recognition, the templates of the input biometric data are compared against the templates in the database to find a match. Multimodal biometric systems offer several advantages over unimodal systems, such as increased accuracy, robustness, and spoofing detection, and are increasingly used in various applications, including identity verification, access control, and forensic analysis.

5.6 SUMMARY

The implementation of the system is explained in chapter 5. This chapter discussed the proposed methodology and steps of implementation. Here, the current chapter provided a detailed explanation of the steps in the proposed approach and how they are implemented.Therefore, the next chapter will discuss the type of testing and test cases of the project.

CHAPTER 6

RESULTS AND DISCUSSION

6.1 PREAMBLE

The previous chapter deals with the implementation of the proposed approach. It consists ofthe proposed approach and various steps required implementing the proposed approach. It also gives a brief view of the implementation of the system using various steps and proposed approaches. This chapter discusses the experimental result of the proposed approach.

6.2 RESULTS

In this section, the results obtained by experimenting with the proposed approach are introduced. The dataset consists of Facial Images, Finger prints and finger Vein. The dataset consists of images which we see in our daily lives. As a part of the results, evaluating the model is also an important part in the model development process. It is very helpful to find the best model that represents our data and how well the chosen model will work in the future. The results are as follows:

From the above figure we can see that an image of some person this face images are usedin the authentication time.

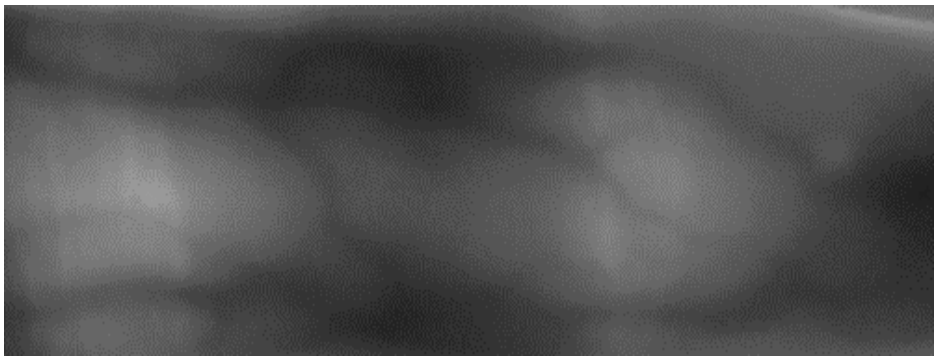


Figure 6.1: Image of Finger Vein

This is the picture of Finger Vein of above fig 1 persons. This Finger Veins are used asBiometric this days to increase the privacy and protection.

Figure 6.2: Image of finger print

The above shown Finger print is belongs to the Fig 6.1 persons. The finger prints are widely usedbiometric authentications. This finger prints are in various ranges different from everyone unique feat

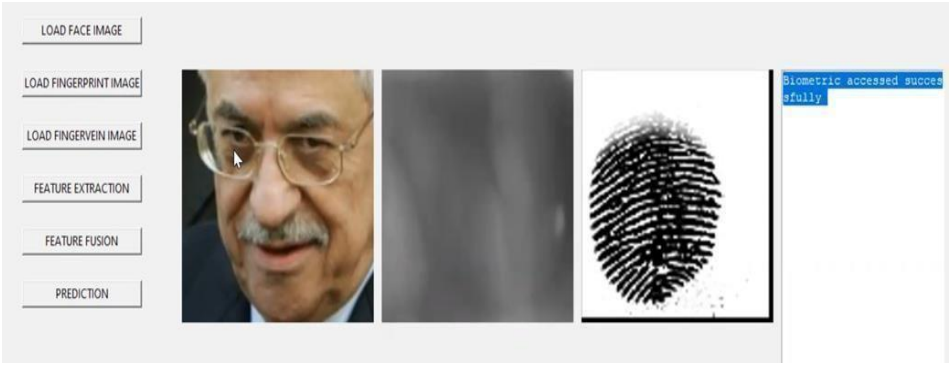


Figure 6.4: Output of Prediction

In above image we can see that in 1st step we are loaded the face image, 2nd we loaded the Finger Print image and lastly we loaded the Finger Vein image. This 3 images are the datasets we are given for prediction. After that this data sets are extracted by the Feature Extraction Technique.

Later on their will be a Feature Fusion this is used to combine the features of the all 3 biometrics whichgiven as authentication data sets already stored in the Database. Now the result we can see that the 3 features are matched well so the result is said to be as Biometrics success successfully

Figure 6.5: Output of the prediction

In above picture 6.4 we can see that when it asked for load face image, here we loaded a something oddimage which is not related to the face now we are given for prediction the result is access denied. We cansee clearly that accuracy of the prediction.

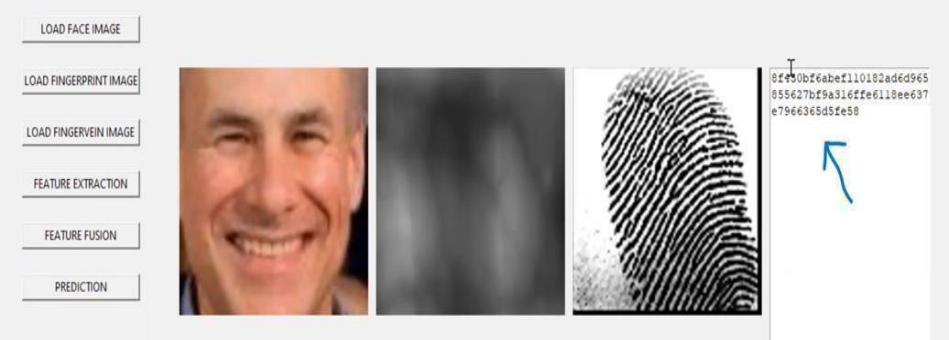


Figure 6.6: Hash key Generation

In above picture 6.5 we can see the how the hash key generated and that hash key is nothing butthe unique key generated everything in the authentication section. This hash is used as the privacypreserving of biometric templated which is used to protect from the reconstruction the template.

This hash key should be match in the end otherwise the access get denied.

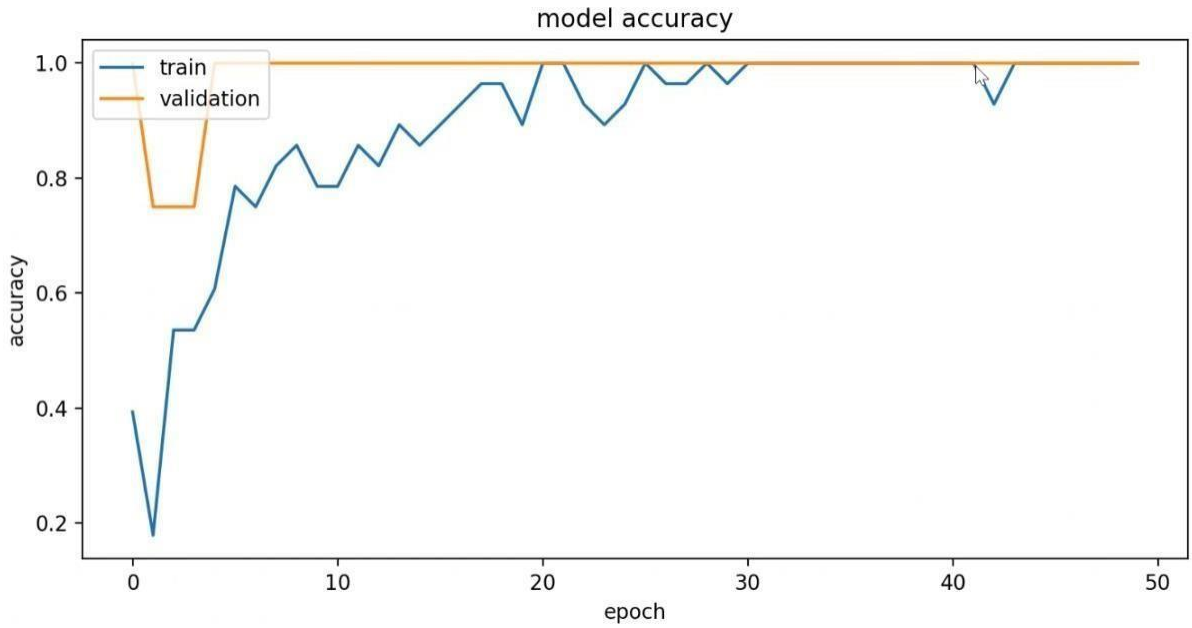


Fig 6.7: Result of Model Accuracy

The above figure 6.6 is used to understand the result of the Model Accuracy of our prosed system which isgenerated by the python code. Here blue line is the training accuracy. We can see how the trained accuracyis increased. The orange line is used to understand the validation that means processing of given data. Thereis a standard accurate

rate.

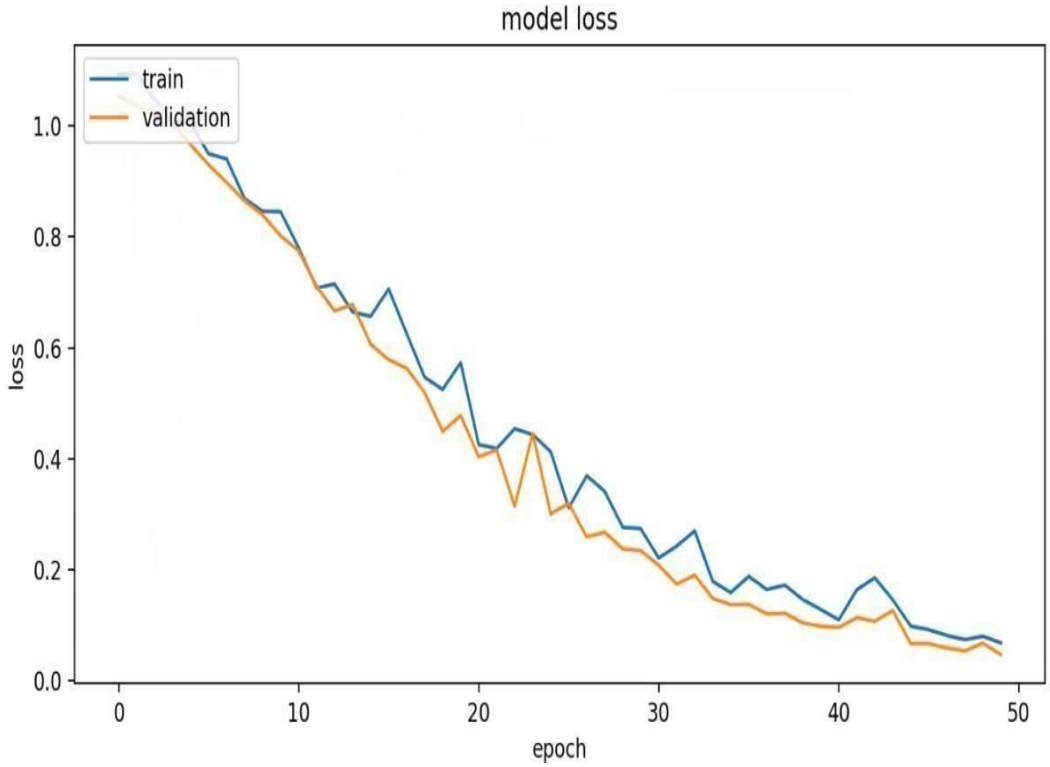


Fig 6.8 : Result of Model loss

Typically, the goal is to minimize the validation loss since this indicates that the model is generalizing well and will perform well on new, unseen data. If the validation loss is much higher than the training loss, it could indicate that the model is overfitting to the training data and is not generalizing well. In thiscase, regularization techniques such as dropout or weight decay can be used to improve generalization.

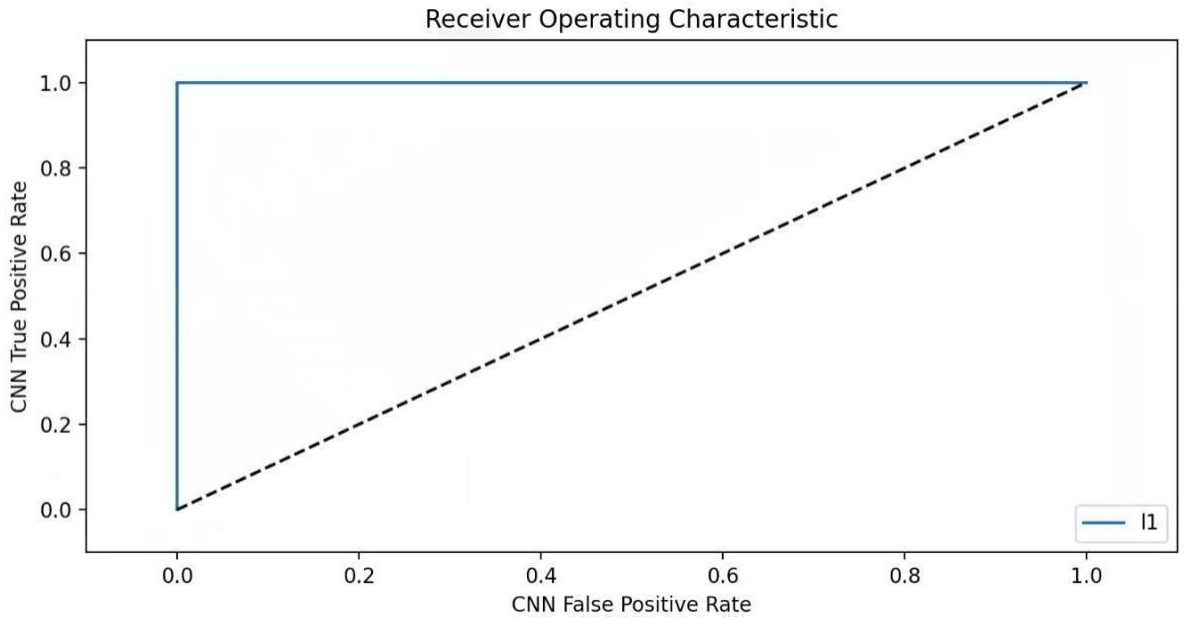


Fig 6.9: Result of True/False Positive Rate

True positive rate (TPR), also known as sensitivity or recall, is the proportion of actual positive cases that arecorrectly identified as positive by the model. It is calculated as:

TPR = TP / (TP + FN)

where TP is the number of true positives (i.e., actual positive cases correctly identified as positive) and FN is thenumber of false negatives (i.e., actual positive cases incorrectly identified as negative).

False positive rate (FPR) is the proportion of actual negative cases that are incorrectly identified as positive by themodel. It is calculated as:

FPR = FP / (FP + TN)

where FP is the number of false positives (i.e., actual negative cases incorrectly identified as positive) and TN is thenumber of true negatives (i.e., actual negative cases correctly identified as negative).

6.2 DISCUSSIONS

The results of template protection using hashing depend on a variety of factors, includingthe quality of the biometric data, the choice of hash function, and the overall security of the system. In general, however, template protection using hashing can providea high level of security and privacy for biometric data. Template protection using hashingis a common method for protectingbiometric data. The basic idea is to use a cryptographic hash function to transform a biometric template into a fixed-length string of

bits, or hash, that can be stored securely without revealing theoriginal template. The hash can then be used for biometric matching without ever exposing the original biometric template.

6.3 SUMMARY

This chapter discusses the experimental results of the proposed work and also explains howobjects gets detected and how the audio feedback of the detected images occurs.

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENT

7.1 CONCLUSION

This study provides a survey on Privacy preserving in face identification techniques based on privacy preserving binning, enrollment and hash generation and many other machine approaches. Through this survey, it is identified that some parameters are not considered anywherein the existing system and it is very much essential for any template protection scheme. Computational complexity and speed of the model, ability to work on hardware is also a challenge which needs to be further addressed.

7.2 FUTURE ENHANCEMENT

The matching rate is very high which we are going to provide in our project. In previous papers theexhaustive search is moderate so we are going to increase the rate of exhaustive search. Additionally, our proposal performs admirably on unrestricted databases, such as the face database and increasesthe resistance of spoofing attacks so that efficiency of the system will increase. Sometimes by using multi authentication if any one biometric is stolen then they can rebuild our features and hack our personal data.

REFERENCES

[1] Z. H. Goh *et al.*, "A Framework for Multimodal Biometric Authentication Systems With Template Protection," in *IEEE Access*, vol. 10, pp. 96388-96402, 2022,doi:10.1109/ACCESS.2022.3205413.

[2] H. Lee, C. Y. Low and A. B. Jin Teoh, "Softmax Out Transformation-Permutation Network for Facial Template Protection," *2020 25th International Conference on Pattern Recognition (ICPR)*,Milan, Italy, 2021, pp. 7558-7565, doi:10.1109/ICPR48806.2021.9413163.

[3] J. Dong, X. Meng, M. Chen and Z. Wang, "Template protection based on DNA coding for multimodal biometric recognition," *2017 4th International Conference on Systemsand Informatics (ICSAI)*, Hangzhou, China, 2017, pp. 1738-1742, doi: 10.1109/ICSAI.2017.8248565.

[4] L. Ghammam, M. Barbier and C. Rosenberger, "Enhancing the Security ofTransformation Based Biometric Template Protection Schemes," *2018 International Conference on Cyberworlds (CW)*, Singapore, 2018, pp. 316-323, doi: 10.1109/CW.2018.00065.

[5] S. Kirchgasser, C. Kauba, Y. -L. Lai, J. Zhe and A. Uhl, "Finger Vein Template Protection Based on Alignment-Robust Feature Description and Index-of-Maximum Hashing," in *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol.2, no. 4, pp. 337-349, Oct. 2020, doi: 10.1109/TBIOM.2020.2981673.

[6] Y. C. Feng, P. C. Yuen and A. K. Jain, "A Hybrid Approach for Generating Secureand Discriminating Face Template," in *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 1, pp. 103-117, March 2010, doi: 10.1109/TIFS.2009.2038760.

[7] Krivokuća Hahn and S. Marcel, "Biometric Template Protection for Neural-Network-Based Face Recognition Systems: A Survey of Methods andEvaluation Techniques," in IEEE Transactions on Information Forensics and

Security, vol. 18, pp. 639-666, 2023, doi: 10.1109/TIFS.2022.3228494.

[8] J. Cui and A. B. J. Te oh, "Deep Index-of-Maximum Hashing for Face Template Protection," 2020 5th International Conference on Computer and Communication Systems (ICCCS), Shanghai, China, 2020, pp. 413-418, doi:10.1109/ICCCS49078.2020.9118594.

[9] T. M. Dang, L. Tran, T. D. Nguyen and D. Choi, "FEHash: Full Entropy Hash forFace Template Protection," 2020 IEEE/CVF Conference on Computer Vision and Pattern

Recognition Workshops (CVPRW), Seattle, WA, USA, 2020, pp. 3527-3536, doi: 10.1109/CVPRW50498.2020.00413.

[10] Jegede, Abayomi et al. "Face Recognition and Template Protection with Shielding Function." International journal of security and its applications 9 (2015): 149-164.

[11] D. Osorio-Roig, C. Rathgeb, P. Drozdowski and C. Busch, "Stable Hash Generation for Efficient Privacy-Preserving Face Identification," in IEEE Transactions on Biometrics, Behavior, and Identity Science, vol. 4, no. 3, pp. 333-348, July 2022, doi: 10.1109/TBIOM.2021.3100639.

[12] A. K. Jindal, S. Chalamala and S. K. Jami, "Face Template Protection Using Deep Convolutional Neural Network," 2018 IEEE/CVF Conference on Computer Vision andPattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA,2018, pp. 575-5758, doi: 10.1109/CVPRW.2018.00087.

[13] P. Bauspieß, J. Kolberg, P. Drozdowski, C. Rathgeb and C. Busch, "Privacy-Preserving Preselection for Protected Biometric Identification UsingPublic-Key Encryption with Keyword Search," in IEEE Transactions on Industrial Informatics,2022, doi: 10.1109/TII.2022.3199944.

[14]Y. -L. Pan, J. -C. Chen and J. -L. Wu, "A Multi-Factor Combinations Enhanced Reversible Privacy Protection System for Facial Images," 2021 IEEE International Conference on Multimedia and Expo (ICME), 2021, pp. 1-6, doi:10.1109/ICME51207.2021.9428264.

[15] L. Weng, L. Amsaleg, A. Morton and S. Marchand Maillet, "A Privacy- Preserving Framework for Large-Scale Content- based Information Retrieval," in IEEE Transactions on Information Forensics and Security, vol. 10, no. 1, pp. 152- 167, Jan. 2015, doi: 10.1109/TIFS.2014.2365998.

[16] Y. Wang, J. Wan, J. Guo, Y. -M. Cheung and P. C. Yuen, "Inference-Based Similarity Search in Randomized Montgomery Domains for Privacy-Preserving Biometric Identification," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 7, pp. 1611-1624, 1 July 2018, doi:10.1109/TPAMI.2017.2727048.

[17] L. Debiasi, S. Kirchgasser, B. Prommegger, A. Uhl, A. Grudzień and M. Kowalski, "Biometric Template Protection in the Image Domain Using Non-invertible Gray-scale Transforms," 2019 IEEE International Workshop on Information Forensics and Security (WIFS), 2019, pp. 1-6,doi:10.1109/WIFS47025.2019.9034984.

[18] W. Xue, W. Hu, P. Gauranvaram, A. Seneviratne and S. Jha, "An Efficient Privacy-preserving IoT System for Face Recognition," 2020 Workshop on Emerging Technologies for Security in IoT (ETSecIoT), 2020, pp. 7-11, doi: 10.1109/ETSecIoT50046.2020.00006

[19] J. Zhang, W. Shen, L. Liu and Z. Wu, "Face recognition model based on privacyprotection and random forest algorithm," 2018 27th Wireless and Optical Communication Conference (WOCC), 2018, pp. 1-5, doi: 10.1109/WOCC.2018.8372707.

[20] O. Sarwar, B. Rinner and A. Cavallaro, "A Privacy Preserving Filter for ObliqueFace Images Based on Adaptive Hopping Gaussian Mixtures," in IEEE Access, vol. 7, pp. 142623-142639, 2019, doi:10.1109/ACCESS.2019.2944861.

[21] R. K. Pandey, Y. Zhou, B. U. Kota and V. Govindaraju, "Deep Secure Encoding for Face Template Protection," 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Las Vegas, NV, USA, 2016, pp. 77-83, doi: 10.1109/CVPRW.2016.17.

[22] J. Yang, J. Liu and J. Wu, "Facial Image Privacy Protection Based on Principal Components of Adversarial Segmented Image Blocks," in IEEE Access, vol. 8, pp. 103385-103394, 2020, doi:10.1109/ACCESS.2020.2999449.

[23] P. Drozdowski, F. Stockhardt, C. Rathgeb, D. Osorio-Roig and C. Busch,"Feature Fusion Methods for Indexing and Retrieval of Biometric Data: Application to Face

Recognition With Privacy Protection,"in IEEE Access, vol. 9, pp. 139361- 139378, 2021, doi: 10.1109/ACCESS.2021.3118830.

[24] G. Mai, K. Cao, X. Lan and P. C. Yuen, "SecureFace: Face Template Protection," in IEEE Transactions on Information Forensics and Security, vol. 16, pp. 262- 277, 2021, doi:10.1109/TIFS.2020.3009590.

[25] M. Gomez-Barrero, J. Galbally, C. Rathgeb and C. Busch, "General Frameworkto Evaluate Unlinkability in Biometric Template Protection Systems," in IEEE Transactionson Information Forensics and Security, vol. 13, no. 6, pp. 1406- 1420, June 2018, doi: 10.1109/TIFS.2017.2788000.

[26] B. Meden et al., "Privacy–Enhancing Face Biometrics: A Comprehensive Survey," in IEEE Transactions on Information Forensics and Security, vol. 16, pp. 4147-4183, 2021, doi:10.1109/TIFS.2021.3096024.

[27] M. Gudavalli, S. V. Raju, A. V. Babu and D. S. Kumar, "Multimodal Biometrics

-- Sources, Architecture and Fusion Techniques: An Overview," 2012 International Symposium on Biometrics and Security Technologies, Taipei, Taiwan, 2012, pp. 27-34, doi: 10.1109/ISBAST.2012.24.

[28] D. H. Davies, S. Ray, M. Gurkowski and L. Lee, "A Biometric Access PersonalOptical Storage Device," 2006 Optical Data Storage Topical Meeting, Montreal, QC, Canada,2006, pp. 100-102, doi: 10.1109/ODS.2006.1632732.

[29] R. Li, D. Tang, B. Huang and W. Li, "How Many Samples Does Convincible Performance Evaluation of a Biometric System Need?," 2012 International Symposium on Biometrics and Security Technologies, Taipei, Taiwan, 2012, pp. 23-26, doi: 10.1109/ISBAST.2012.9.

[30] H. Ni, D. Li, T. Isshiki and H. Kunieda, "Robust Multiple Minutiae Partitions forFingerprint Authentication," 2012 International Symposium on Biometrics and Security Technologies, Taipei, Taiwan, 2012, pp. 35-44, doi: 10.1109/ISBAST.2012.19.

[31] Information Forensics A. Z. Bendale and T. E. Boult, "id-Privacy in large scale biometric systems," 2010 IEEE International Workshop on and Security, Seattle, WA, USA,2010, pp. 1-6, doi: 10.1109/WIFS.2010.5711439.

[32] L. Ghammam, M. Barbier and C. Rosenberger, "Enhancing the Security of Transformation Based Biometric Template Protection Schemes," 2018 International

Conference on Cyberworlds (CW), Singapore, 2018, pp. 316-323, doi: 10.1109/CW.2018.00065.

[33] J. Cui and A. B. J. Teoh, "Deep Index-of-Maximum Hashing for Face TemplateProtection," 2020 5th International Conference on Computer and Communication Systems (ICCCS), Shanghai, China, 2020, pp. 413-418, doi: 10.1109/ICCCS49078.2020.9118594.

[34] W. Ponce-Hernandez, R. Blanco-Gonzalo, R. Sanchez-Reillo and J. Liu- Jimenez, "Template protection approaches: Fuzzy Vault scheme," 2019 International Carnahan Conference on Security Technology (ICCST), Chennai, India,2019, pp. 1-5, doi: 10.1109/CCST.2019.8888405.

[35] W. Ponce-Hernandez, R. Blanco-Gonzalo, R. Sanchez-Reillo and J. Liu- Jimenez, "Template protection approaches: Fuzzy Vault scheme," 2019 International Carnahan Conference on Security Technology (ICCST), Chennai, India,2019, pp. 1-5, doi: 10.1109/CCST.2019.8888405.

[36] W. Huang, X. Yu, Q. Li and X. Niu, "A New Face Recognition Scheme with Renewable Templates," 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Harbin, China, 2008, pp. 1426-1429, doi: 10.1109/IIH-MSP.2008.212.

[37] Q. Wang, W. Huang, X. Niu and X. Jiang, "A Template Protection Scheme For Statistic Feature-based 2D Face Recognition," 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Harbin, China, 2008,pp. 374-377, doi: 10.1109/IIH-MSP.2008.223.

[38] S. Kirchgasser et al., "Template Protection on Multiple Facial Biometrics in theSignal Domain under Visible and Near-Infrared Light," 2020 8th International Workshop on Biometrics and Forensics (IWBF), Porto, Portugal, 2020, pp. 1-6, doi:10.1109/IWBF49977.2020.9107964.

[39] M. A. Dabbah, W. L. Woo and S. S. Dlay, "Appearance-Based Biometric Recognition: Secure Authentication and Cancellability," 2007 15th International Conference on Digital Signal Processing, Cardiff, UK, 2007, pp. 479-482, doi: 10.1109/ICDSP.2007.4288623.

[40] Z. H. Goh et al., "A Framework for Multimodal BiometricAuthenticationSystems With Template Protection," in IEEE Access, vol. 10, pp. 96388-96402,2022, doi: 10.1109/ACCESS.2022.3205413.

[41] H. Lee, C. Y. Low and A. B. Jin Teoh, "SoftmaxOut Transformation-Permutation Network for Facial Template Protection," 2020 25thInternational Conference on Pattern Recognition (ICPR), Milan, Italy, 2021, pp.7558-7565, doi: 10.1109/ICPR48806.2021.9413163.

[42] J. Dong, X. Meng, M. Chen and Z. Wang, "Template protection basedon DNA coding for multimodal

biometric recognition," 2017 4th International Conference on Systems and Informatics (ICSAI), Hangzhou, China, 2017, pp. 1738-1742, doi: 10.1109/ICSAI.2017.8248565.

[43] Y. C. Feng, P. C. Yuen and A. K. Jain, "A Hybrid Approach for Generating Secure and Discriminating Face Template," in IEEE Transactions on Information Forensics and Security, vol. 5, no. 1, pp. 103-117, March 2010, doi: 10.1109/TIFS.2009.2038760.

[44] D. Osorio-Roig, C. Rathgeb, P. Drozdowski and C. Busch, "Stable Hash Generation for Efficient Privacy-Preserving Face Identification," in IEEE Transactions on Biometrics, Behavior, and Identity Science, vol. 4, no. 3, pp. 333-348, July 2022, doi: 10.1109/TBIOM.2021.3100639.

[45] Y. C. Feng and P. C. Yuen, "Binary Discriminant Analysis for Face Template Protection," 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 2010, pp. 874-877, doi: 10.1109/ICPR.2010.220.

APPENDIX A

SOURCE CODE

```
import cv2
import numpy as npimport os

from matplotlib import pyplot as plt
#from multimodal_face_and_fp_project import
extract_feattfrom skimage.color import rgb2gray

import pickle
from tkinter import messagebox from PIL import
ImageTk, Imageimport tensorflow as tf

from tensorflow.keras.preprocessing.image import
ImageDataGeneratorfrom tensorflow.keras.preprocessing
import image

from tensorflow.keras.applications.vgg19 import
VGG19

from tensorflow.keras.applications.vgg19 import
preprocess_input#from keras.layers import Flatten

from keras.layers import merge, Inputimport h5py

from tensorflow.keras.layers import Dense,
Activation, Flattenimage_input = Input(shape=(160,160,3))

weight_path = "Multi"

model =
```

```
VGG19(weights="vgg19_weights_tf_dim_ordering_t
f_kernels_notop.h5",include
_top=False)

from sklearn.datasets import load_filesfrom
keras.utils import np_utils

import numpy as np from glob import glob tar=3
path='./Fingervein_data/'

# define function to load train, test, and validation
datasetsdef load_dataset(path):

data = load_files(path)

files = np.array(data['filenames'])

targets =
np_utils.to_categorical(np.array(data['target']), tar)return
files, targets

# load train, test, and validation datasets train_files,
train_targets = load_dataset(path)test_files=train_files

test_targets = train_targets# get the burn classes

# We only take the characters from a starting position
to remove the path#burn_classes = [item[11:-1] for item in
sorted(glob(path))]

burn_classes = [item[10:-1] for item in
sorted(glob("./Fingervein_data/*/"))]# print statistics about
the dataset
```

```
        print('There are %d total categories.' %
len(burn_classes))

        print(burn_classes)

        print('There are %s total images.\n' %
len(np.hstack([train_files, test_files])))print('There are %d
training images.' % len(train_files))

        print('There are %d test images.'% len(test_files)) for
file in train_files: assert('.DS_Store' not in file from
tensorflow.keras.preprocessing import imagefrom tqdm
import tqdm

        # Note: modified these two functions, so that we can
later also read the inceptiontensors which

        # have a different format

        def path_to_tensor(img_path, width=224,
height=224):# loads RGB image as PIL.Image.Image type

        img = image.load_img(img_path, target_size=(width,
height))

        # convert PIL.Image.Image type to 3D tensor with
shape (width, heigth, 3)x = image.img_to_array(img)

        # convert 3D tensor to 4D tensor with shape (1,
width, height, 3) and return 4Dtensor

        return np.expand_dims(x, axis=0)
        def paths_to_tensor(img_paths, width=224,
height=224):
        list_of_tensors = [path_to_tensor(img_path, width,
height) for img_path intqdm(img_paths)]
```

```
        return np.vstack(list_of_tensors)


        import kerasimport timeit


        # callback to show the total time taken during training
and for each epochclass
EpochTimer(keras.callbacks.Callback):
        train_start = 0

          train_end = 0
        epoch_start = 0
        epoch_end =  0 def get_time(self):
        return  timeit.default_timer() def on_train_begin(self,
logs={}):
        self.train_start = self.get_time()def on_train_end(self,
logs={}):
        self.train_end = self.get_time()
        print('Training took {} seconds'.format(self.train_end
- self.train_start))def on_epoch_begin(self, epoch, logs={}):
        self.epoch_start = self.get_time()
        def on_epoch_end(self, epoch, logs={}):
self.epoch_end = self.get_time()
        print('Epoch                          {}
    took   {}   seconds'.format(epoch,   self.epoch_end   -
self.epoch_start))


        from PIL import ImageFile
ImageFile.LOAD_TRUNCATED_IMAGES = True# pre-
process the data for Keras
        train_tensors =
```

```
paths_to_tensor(train_files).astype('float32')/255test_tensors
= paths_to_tensor(test_files).astype('float32')/255

    from tensorflow.keras.layers import Conv2D,
    MaxPooling2D,
    GlobalAveragePooling2D
    from tensorflow.keras.layers import Dropout, Flatten,
Densefrom tensorflow.keras.models import Sequential

    from tensorflow.keras.models import Model
    from tensorflow.keras.callbacks import
ModelCheckpointimport matplotlib.pyplot as plt

    img_width, img_height = 224, 224
    batch_size = 8epoch=50


    ########
    img_width, img_height = img_width, img_height
batch_size = 32

    samples_per_epoch = 10
    validation_steps = 300
    nb_filters1 = 32
    nb_filters2 = 64
    conv1_size = 3
    conv2_size = 3
    pool_size = 3
    lr = 0.0004
    from tensorflow.keras import optimizers
    from tensorflow.keras.models import Sequential
    from tensorflow.keras.layers import Dropout, Flatten,
Dense, Activationfrom tensorflow.keras.layers import
Convolution2D, MaxPooling2D from tensorflow.keras
import callbacks

    import time
    #input_shape=(img_width, img_height,3)model =
Sequential()
```

```
        model.add(Convolution2D(nb_filters1, conv1_size,
conv1_size, padding='same',input_shape=(img_width,
img_height, 3)))
        model.add(Activation("relu"))
model.add(MaxPooling2D(pool_size=(pool_size, pool_size)))
        model.add(Convolution2D(nb_filters2, conv2_size,
conv2_size, padding='same'))model.add(Activation("relu"))
        model.add(MaxPooling2D(pool_size=(pool_size,
pool_size)))model.add(Flatten())
        model.add(Dense(256))
model.add(Activation("relu"))
model.add(Dropout(0.5))model.add(Dense(tar,
activation='softmax'))
model.compile(loss='categorical_crossentropy',
        optimizer=optimizers.RMSprop(lr=lr),
metrics=['accuracy'])
        hist=model.fit(train_tensors, train_targets
,validation_split=0.1, epochs=epoch,batch_size=64)
        #model.save('color_trained_modelDNN.h5')
model.save('vein_CNN.h5')
        ##############
        path='./Fingerprint_data/'
        # define function to load train, test, and validation
datasetsdef load_dataset(path):
        data = load_files(path)
        files = np.array(data['filenames'])
        targets =
np_utils.to_categorical(np.array(data['target']), tar)return
files, targets
        # load train, test, and validation datasets train_files,
train_targets = load_dataset(path)
        test_files=train_files test_targets = train_targets
```

```
        # get the burn classes
        # We only take the characters from a starting position
    to remove the path#burn_classes = [item[11:-1] for item in
    sorted(glob(path))]
        burn_classes = [item[10:-1] for item in
    sorted(glob("./Fingerprint_data/*/"))]# print statistics about
    the dataset
        print('There are %d total categories.' %
    len(burn_classes))
print(burn_classes)
        print('There are %s total images.\n' %
    len(np.hstack([train_files, test_files])))print('There are %d
    training images.' % len(train_files))
        print('There are %d test images.'% len(test_files))
        for file in train_files: assert('.DS_Store' not in file)
        from tensorflow.keras.preprocessing import image
    from tqdm import tqdm
        # Note: modified these two functions, so that we can
    later also read the inceptiontensors which
        # have a different format
        def path_to_tensor(img_path, width=224,
    height=224):# loads RGB image as PIL.Image.Image type
        img = image.load_img(img_path, target_size=(width,
    height))
        # convert PIL.Image.Image type to 3D tensor with
    shape (width, heigth, 3)x = image.img_to_array(img)
        # convert 3D tensor to 4D tensor with shape (1,
    width, height, 3) and return 4Dtensor
        return np.expand_dims(x, axis=0)
        def paths_to_tensor(img_paths, width=224,
    height=224):
        list_of_tensors = [path_to_tensor(img_path, width,
    height) for img_path intqdm(img_paths)]
        return np.vstack(list_of_tensors)
```

```
import kerasimport timeit

# callback to show the total time taken during training
and for each epochclass
EpochTimer(keras.callbacks.Callback):

train_start = 0
train_end = 0
epoch_start = 0
epoch_end = 0 def get_time(self):

return timeit.default_timer()



def on_train_begin(self, logs={}):self.train_start = self.get_time()

def on_train_end(self, logs={}): self.train_end = self.get_time()

print('Training took {} seconds'.format(self.train_end - self.train_start))def
on_epoch_begin(self, epoch, logs={}):

self.epoch_start = self.get_time()
def on_epoch_end(self, epoch, logs={}):self.epoch_end = self.get_time()

print('Epoch {} took {} seconds'.format(epoch, self.epoch_end -
self.epoch_start))
from PIL import ImageFile ImageFile.LOAD_TRUNCATED_IMAGES = True



# pre-process the data for Keras
train_tensors = paths_to_tensor(train_files).astype('float32')/255test_tensors =
paths_to_tensor(test_files).astype('float32')/255

from tensorflow.keras.layers import Conv2D,
MaxPooling2D,
GlobalAveragePooling2D
from tensorflow.keras.layers import Dropout, Flatten, Densefrom tensorflow.keras.models
import Sequential

from tensorflow.keras.models import Model
from tensorflow.keras.callbacks import ModelCheckpointimport matplotlib.pyplot as plt
```

```
img_width, img_height = 224, 224
batch_size = 8
########
img_width, img_height = img_width, img_heightbatch_size = 32

samples_per_epoch = 10
validation_steps = 300
nb_filters1 = 32
nb_filters2 = 64
conv1_size = 3
conv2_size = 3
pool_size = 3
lr = 0.0004
from tensorflow.keras import optimizers
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dropout, Flatten, Dense, Activationfrom
tensorflow.keras.layers import Convolution2D, MaxPooling2D from tensorflow.keras import
callbacks

import time
#input_shape=(img_width, img_height,3)model = Sequential()

model.add(Convolution2D(nb_filters1, conv1_size, conv1_size, padding='same',
input_shape=(img_width, img_height, 3)))
model.add(Activation("relu")) model.add(MaxPooling2D(pool_size=(pool_size,
pool_size)))

model.add(Convolution2D(nb_filters2, conv2_size, conv2_size, padding='same'))
model.add(Activation("relu"))

model.add(MaxPooling2D(pool_size=(pool_size, pool_size)))model.add(Flatten())

model.add(Dense(256)) model.add(Activation("relu")) model.add(Dropout(0.5))
model.add(Dense(tar, activation='softmax'))

model.compile(loss='categorical_crossentropy', optimizer=optimizers.RMSprop(lr=lr),
metrics=['accuracy'])

hist=model.fit(train_tensors, train_targets ,validation_split=0.1, epochs=epoch,
batch_size=64)
#model.save('color_trained_modelDNN.h5')model.save('fingerprint_CNN.h5')
```

```
#############
path='./Face_data/'
# define function to load train, test, and validation datasetsdef load_dataset(path):

data = load_files(path)
files = np.array(data['filenames'])
targets = np_utils.to_categorical(np.array(data['target']), tar)return files, targets

# load train, test, and validation datasets train_files, train_targets = load_dataset(path)

test_files=train_files test_targets = train_targets

# get the burn classes
# We only take the characters from a starting position to remove the path
#burn_classes = [item[11:-1] for item in sorted(glob(path))] burn_classes = [item[10:-1] for item in sorted(glob("./Face_data/*/"))]# print statistics about the dataset

print('There are %d total categories.' % len(burn_classes))print(burn_classes)

print('There are %s total images.\n' % len(np.hstack([train_files, test_files])))print('There are %d training images.' % len(train_files))

print('There are %d test images.'% len(test_files)) for file in train_files: assert('.DS_Store' not in file)from tensorflow.keras.preprocessing import imagefrom tqdm import tqdm

# Note: modified these two functions, so that we can later also read the inceptiontensors which
# have a different format
def path_to_tensor(img_path, width=224, height=224):# loads RGB image as PIL.Image.Image type

img = image.load_img(img_path, target_size=(width, height))
# convert PIL.Image.Image type to 3D tensor with shape (width, heigth, 3)x = image.img_to_array(img)

# convert 3D tensor to 4D tensor with shape (1, width, height, 3) and return 4Dtensor
return np.expand_dims(x, axis=0)
def paths_to_tensor(img_paths, width=224, height=224):
list_of_tensors = [path_to_tensor(img_path, width, height) for img_path in tqdm(img_paths)]
return np.vstack(list_of_tensors)
import kerasimport timeit
```

```
# graph the history of model.fit def show_history_graph(history):

# summarize history for accuracy plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])plt.title('model accuracy') plt.ylabel('accuracy')
plt.xlabel('epoch')

plt.legend(['train', 'validation'], loc='upper left')plt.show()

# summarize history for loss plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])plt.title('model loss') plt.ylabel('loss') plt.xlabel('epoch')

plt.legend(['train', 'validation'], loc='upper left')plt.show()


# callback to show the total time taken during training and for each epochclass
EpochTimer(keras.callbacks.Callback):

train_start = 0
train_end = 0
epoch_start = 0
epoch_end = 0


def get_time(self):
return  timeit.default_timer() def on_train_begin(self, logs={}):

self.train_start = self.get_time()def on_train_end(self, logs={}):

self.train_end = self.get_time()
print('Training took {} seconds'.format(self.train_end - self.train_start))def
on_epoch_begin(self, epoch, logs={}):

    self.epoch_start = self.get_time()
def on_epoch_end(self, epoch, logs={}):self.epoch_end = self.get_time()

print('Epoch {} took {} seconds'.format(epoch, self.epoch_end -
self.epoch_start))
from PIL import ImageFile ImageFile.LOAD_TRUNCATED_IMAGES = True

# pre-process the data for Keras
train_tensors = paths_to_tensor(train_files).astype('float32')/255test_tensors =
paths_to_tensor(test_files).astype('float32')/255
```

```
from tensorflow.keras.layers import Conv2D, MaxPooling2D,
GlobalAveragePooling2D
from tensorflow.keras.layers import Dropout, Flatten, Densefrom tensorflow.keras.models
import Sequential

from tensorflow.keras.models import Model
from tensorflow.keras.callbacks import ModelCheckpointimport matplotlib.pyplot as plt

img_width, img_height = 224, 224
batch_size = 8
########
img_width, img_height = img_width, img_heightbatch_size = 32

samples_per_epoch = 10
validation_steps = 300
nb_filters1 = 32
nb_filters2 = 64
conv1_size = 3
conv2_size = 3
pool_size = 3
lr = 0.0004
from tensorflow.keras import optimizers
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dropout, Flatten, Dense, Activationfrom
tensorflow.keras.layers import Convolution2D, MaxPooling2D from tensorflow.keras import
callbacks

import time
#input_shape=(img_width, img_height,3)model = Sequential()

model.add(Convolution2D(nb_filters1, conv1_size, conv1_size, padding='same',
input_shape=(img_width, img_height, 3)))
model.add(Activation("relu")) model.add(MaxPooling2D(pool_size=(pool_size,
pool_size)))

model.add(Convolution2D(nb_filters2, conv2_size, conv2_size, padding='same'))
model.add(Activation("relu"))

model.add(MaxPooling2D(pool_size=(pool_size, pool_size)))model.add(Flatten())

model.add(Dense(256)) model.add(Activation("relu")) model.add(Dropout(0.5))
```

```
model.add(Dense(tar, activation='softmax'))

    model.compile(loss='categorical_crossentropy', optimizer=optimizers.RMSprop(lr=lr),
metrics=['accuracy'])

    hist=model.fit(train_tensors, train_targets ,validation_split=0.2, epochs=epoch,
batch_size=64)
    show_history_graph(hist)
    test_loss, test_acc = model.evaluate(train_tensors, train_targets)
y_pred=model.predict(train_tensors)

    from sklearn.metrics import confusion_matrix,accuracy_score
    cm =
    confusion_matrix(np.argmax(train_targets, axis=1),np.argmax(y_pred,axis=1))
    from sklearn.metrics import roc_curve
    # Calculate ROC curve from y_test and pred
    fpr, tpr, thresholds = roc_curve(np.argmax(test_targets,
axis=1)>=1,np.argmax(y_pred, axis=1)>=1)
    accuracycnn = accuracy_score(np.argmax(test_targets, axis=1),np.argmax(y_pred,axis=1))
    print("CNN confusion matrics=",cm)print("  ")
    print("CNN accuracy=",accuracycnn*100)



    # Plot the ROC curve
    fig = plt.figure(figsize=(8,8)) plt.title('Receiver Operating Characteristic')



    # Plot ROC curve plt.plot(fpr, tpr, label='l1') plt.legend(loc='lower right')



    # Diagonal 45 degree lineplt.plot([0,1],[0,1],'k--')



    # Axes limits and labelsplt.xlim([-0.1,1.1])
    plt.ylim([-0.1,1.1])
    plt.ylabel('CNN True Positive Rate')
    plt.xlabel('CNN False Positive Rate')
```

```
        plt.show()
#model.save('color_trained_modelDNN.h5') model.save('face_CNN.h5')
model.save('vgg19_weights_tf_dim_ordering_tf_kernel_notop.h5')#############


        image_input = Input(shape=(160,160,3))weight_path = "Multi"
        model                                                    =
        VGG19(weights="vgg19_weights_tf_dim_ordering_tf_kernels_notop.h5",include
        _top=False)
import matplotlib.pyplot as plt
        from sk_dsp_comm.fec_conv import FECConvfrom sk_dsp_comm import digitalcom as dc
import numpy as np

        cc = FECConv()
        def extract_featt(img):
        #img_path= "D:\\Multi\\dataset\\Face 1\\1.png"
        #img = image.load_img(img_path,target_size=(160, 160))img_data =
image.img_to_array(img)

        img_data = np.expand_dims(img_data, axis=0)img_data = preprocess_input(img_data)
features = model.predict(img_data)

        flat_feat = features.flatten()print(flat_feat.shape) return flat_feat

        def resiz(main_img):
        re_face = cv2.resize(main_img,(160,160))mean, std = re_face.mean(), re_face.std() re_face
= (re_face-mean)/std

        re_face = re_face*225


        #cv2.imshow("kgkjv",re_face)return re_face

        face_datas =[]fp_datas =[] fv_datas =[] x=0

        target=[]
        folder_list =os.listdir('Face_data')for folder in folder_list:

        # create a path to the folder path ='Face_data/'+ str(folder)img_files  =  os.listdir(path) for
file in img_files:
```

```
src = os.path.join(path, file) main_img = cv2.imread(src)res=resiz(main_img)

#re_face = cv2.resize(main_img,(160,160))res1=extract_featt(res) face_datas.append(res1)

#ress1 = feature_ex(res)target.append(x)

x=x+1
#..... #preprocess

def resizz(main_img1):
resiz_fp = cv2.resize(main_img1,(160,160))#actual size of fp(160,160)#apply enhancement

enhan = fingerprint_enhancer.enhance_Fingerprint(resiz_fp)cv2.imwrite('pre.png',enhan)
#cv2.imshow("enhamce_img",enhan)

return enhan
folder_list =os.listdir('Fingerprint_data')
for folder in folder_list:
# create a path to the folder
path ='Fingerprint_data/'+ str(folder)img_files = os.listdir(path)


for file in img_files:
src = os.path.join(path, file) main_img1 = cv2.imread(src)#res1 = resizz(main_img1)

main_img1= cv2.resize(main_img1,(160,160))res2=extract_featt(main_img1)
fp_datas.append(res2)

#ress2 = feature_ex(res1) folder_list =os.listdir('Fingervein_data')for folder in folder_list:

# create a path to the folder
path ='Fingervein_data/'+ str(folder)img_files = os.listdir(path)

for file in img_files:
src = os.path.join(path, file) main_img1 = cv2.imread(src)#res1 = resizz(main_img1)

main_img1= cv2.resize(main_img1,(160,160))res2=extract_featt(main_img1)
fv_datas.append(res2)

#ress2 = feature_ex(res1)
##from sklearn.ensemble import RandomForestClassifier##from sklearn.datasets import
make_classification ####xc=[]
```

```
##import warnings ##warnings.filterwarnings("ignore")

X = np.concatenate((face_datas,fp_datas,fv_datas),axis=1)

final_fea=[] import hashlib

for i in range(X.shape[0]):



out=X[i,:] out1=np.zeros((out.shape[0]),)for i in range(out.shape[0]):

if out[i]>150:out1[i]=1

z = cc.viterbi_decoder(out1)z=str(z)

result = hashlib.sha256(z.encode()) final_fea.append(result.hexdigest())

np.save('hashing.npy',final_fea)from tkinter import *

import tkinter as tkimport cv2

import os import pickle

from numpy import save
from keras.utils import np_utilsimport os

from tkinter import filedialogimport cv2

import numpy as npimport os

from matplotlib import pyplot as plt
#from multimodal_face_and_fp_project import extract_feattfrom skimage.color import
rgb2gray

import pickle
from tkinter import messagebox from PIL import ImageTk, Image

import tensorflow as tf
from tensorflow.keras.preprocessing.image import ImageDataGeneratorfrom
tensorflow.keras.preprocessing import image

from tensorflow.keras.applications.vgg19 import VGG19
from tensorflow.keras.applications.vgg19 import preprocess_input#from keras.layers
import Flatten
```

```
from keras.layers import merge, Inputimport h5py

from tensorflow.keras.layers import Dense, Activation, Flattenfrom
tensorflow.keras.models import load_model

model1 = load_model('face_CNN.h5') model2= load_model('fingerprint_CNN.h5')
model3= load_model('vein_CNN.h5') import matplotlib.pyplot as plt

from sk_dsp_comm.fec_conv import FECConvfrom sk_dsp_comm import digitalcom as dc
import numpy as np

cc = FECConv()
image_input = Input(shape=(160,160,3))weight_path = "Multi"

model                                                                    =
VGG19(weights="vgg19_weights_tf_dim_ordering_tf_kernels_notop.h5",include
_top=False)
from tensorflow.keras.preprocessing import imagefrom tqdm import tqdm

# Note: modified these two functions, so that we can later also read the inceptiontensors which
# have a different format
def path_to_tensor(img_path, width=224, height=224):# loads RGB image as
PIL.Image.Image type print(img_path)

img = image.load_img(img_path, target_size=(width, height))
# convert PIL.Image.Image type to 3D tensor with shape (width, heigth, 3)x =
image.img_to_array(img)

# convert 3D tensor to 4D tensor with shape (1, width, height, 3) and return 4Dtensor
return np.expand_dims(x, axis=0)
def paths_to_tensor(img_paths, width=224, height=224): list_of_tensors =
[path_to_tensor(img_paths, width, height)]return np.vstack(list_of_tensors)

def extract_featt(img):
#img_path= "D:\\Multi\\dataset\\Face 1\\1.png"
#img = image.load_img(img_path,target_size=(160, 160))img_data =
image.img_to_array(img)

img_data = np.expand_dims(img_data, axis=0)img_data = preprocess_input(img_data)
features = model.predict(img_data)
```

```
flat_feat = features.flatten()print(flat_feat.shape) return flat_feat

def load_face():
filename = filedialog.askopenfilename(title='open')main_img = cv2.imread(filename)

f_image= cv2.imread(filename) f_image=cv2.resize(image,(250,250))return f_image

def preprocess(f_image): image=cv2.resize(f_image,(250,250)) cv2.imshow('Original
Image',image) mean, std = image.mean(), image.std() image = (image-mean)/std
cv2.imshow('Normalized Image',image)cv2.imwrite('Normalized.jpg', image)

return image
def face_features(image):x=extract_featt(image)z=x

#messagebox.showinfo('Feature Extractted ',z)return z

def load_fp():
filename = filedialog.askopenfilename(title='open')main_img = cv2.imread(filename)

fp_image= cv2.imread(filename) fp_image=cv2.resize(image,(250,250))return fp_image

def preprocess(fp_image): image=cv2.resize(fp_image,(250,250))cv2.imshow('Original
Image',image)

out = fingerprint_enhancer.enhance_Fingerprint(image)cv2.imshow('Enhanced Image',out)
cv2.imwrite("Enhanced.jpg", out)

return out
def fp_features(out): x=extract_featt(out)z1=x

#messagebox.showinfo('Feature Extractted ',z)return z

def resiz(main_img):
re_face = cv2.resize(main_img,(160,160))mean, std = re_face.mean(), re_face.std() re_face
= (re_face-mean)/std

re_face = re_face*225 #cv2.imshow("kgkjv",re_face)return re_face

face_f =[]
fin_f=[]
class Window(Frame):
def _init_(self, master=None):Frame._init_(self, master) self.master = master

# changing the title of our master widget self.master.title("Multimodal Biometrics Using
```

```
Deep Hashing ")

        self.pack(fill=BOTH, expand=1)w = tk.Label(root,

        text="Multimodal Biometrics Using Deep Hashing ",fg = "white",

        bg = "black",

        font = "Helvetica 20 bold italic")

        w.pack() w.place(x=350, y=0)#creating buttons

        quitButton = Button(self,command=self.query, text="LOAD FACE
IMAGE",fg="black",activebackground="light grey",width=20)

        quitButton.place(x=50, y=100)

        quitButton = Button(self,command=self.query1, text="LOAD
FINGERPRINT IMAGE",fg="black",activebackground="light gray",width=20)

        quitButton.place(x=50, y=150)

        quitButton = Button(self,command=self.query2,text="LOAD FINGERVEIN
IMAGE",fg="black",activebackground="light grey",width=20)

        quitButton.place(x=50, y=200)

        quitButton = Button(self,command=self.feature,text="FEATURE
EXTRACTION",fg="black",activebackground="light grey",width=20)

        quitButton.place(x=50, y=250)

        quitButton = Button(self,command=self.fusion,text="FEATURE
FUSION",activebackground="light grey",fg="black",width=20)

        quitButton.place(x=50, y=300)

        quitButton                                                    =

  Button(self,command=self.predict,text="PREDICTION",activebackground="light
grey",fg="black",width=20)

        quitButton.place(x=50, y=350) load = Image.open("gray.bmp")

        render = ImageTk.PhotoImage(load)

        image2=Label(self, image=render,borderwidth=15, highlightthickness=5,height=200,
width=200, bg='white')

        image2.image = render image2.place(x=250, y=150)

        image3=Label(self, image=render,borderwidth=15, highlightthickness=5,height=200,
width=200, bg='white')

        image3.image = render image3.place(x=750, y=150)

        image4=Label(self, image=render,borderwidth=15, highlightthickness=5,height=200,
width=200, bg='white')

        image4.image = render image4.place(x=500, y=150)
```

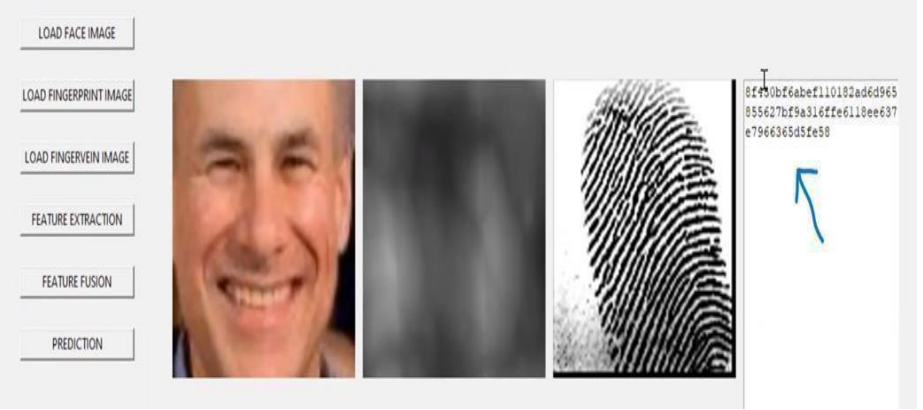
```
#for face_image_load

def query(self, event=None): contents ="Loading Image..."global T,rep

T = Text(self, height=19, width=25)#T.pack()

T.place(x=1000, y=150)T.insert(END,contents) print(contents)

rep = filedialog.askopenfilenames()img = cv2.imread(rep[0]) #cv2.imshow('fff2',img)

img = cv2.resize(img,(250,250))#cv2.imshow('fff1',img)

 Input_img=img.copy()

print(rep[0]) #cv2.imshow('fff',Input_img)

self.from_array = Image.fromarray(cv2.resize(img,(250,250)))load = Image.open(rep[0])

render = ImageTk.PhotoImage(load.resize((250,250)))#cv2.imshow('fff',render)

image2=Label(self,  image=render,borderwidth=15,    highlightthickness=5,height=200, width=200, bg='white')

image2.image = render image2.place(x=250, y=150) contents="Image Loadeded successfully !!T = Text(self, height=19, width=25) #T.pack()

T.place(x=1000, y=150) T.insert(END,contents) print(contents) self.Input_img=Input_img

def close_window(): Window.destroy():

def feature(self, event=None): contents ="Feature Extracting..."global  T,rep,rep1,rep2 main_img = cv2.imread(rep[0]) res=resiz(main_img)

#re_face = cv2.resize(main_img,(160,160)) res1=extract_featt(res) #messagebox.showinfo('Feature Extractted ',z)main_img = cv2.imread(rep1[0])

#re_face = cv2.resize(main_img,(160,160)) res= cv2.resize(main_img,(160,160)) res2=extract_featt(res) #messagebox.showinfo('Feature Extractted ',z)

main_img = cv2.imread(rep2[0]) #re_face = cv2.resize(main_img,(160,160)) res= cv2.resize(main_img,(160,160)) res3=extract_featt(res) #messagebox.showinfo('Feature Extractted ',z)X = np.concatenate((res1,res2,res3))

import hashlibout=X

out1=np.zeros((out.shape[0]),)for i in range(out.shape[0]):
```

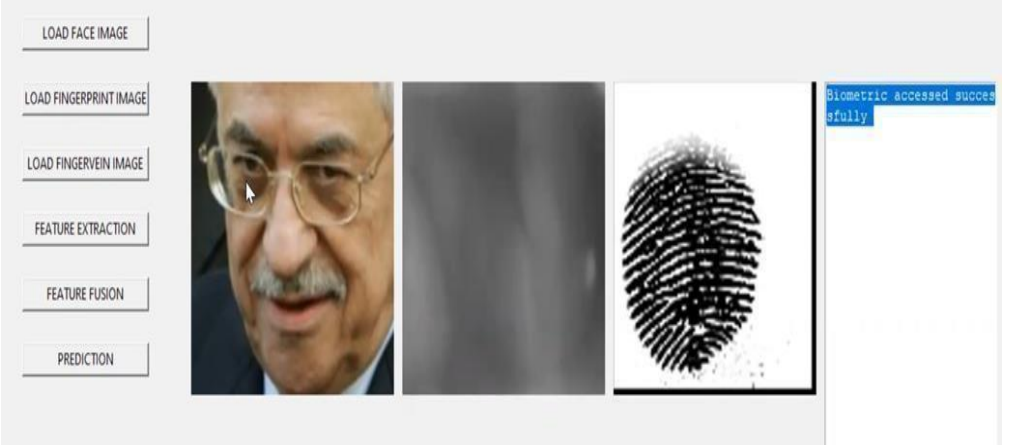
```
if out[i]>150:out1[i]=1

z = cc.viterbi_decoder(out1)z=str(z)

result = hashlib.sha256(z.encode()) final_fea=result.hexdigest() contents=final_fea
np.save('temp_hashing.npy',final_fea)T = Text(self, height=19, width=25) #T.pack()

T.place(x=1000, y=150)T.insert(END,contents) print(contents)

def query1(self, event=None): contents ="Loading Image..."global T,rep1

T = Text(self, height=19, width=25)#T.pack()

T.place(x=1000, y=150)T.insert(END,contents) print(contents)

rep1 = filedialog.askopenfilenames()img = cv2.imread(rep1[0]) #cv2.imshow('fff2',img)

img = cv2.resize(img,(256,256))#cv2.imshow('fff1',img) Input_img=img.copy()
print(rep1[0]) cv2.imwrite('fin.png',img) render = Image.open('fin.png')

render = ImageTk.PhotoImage(render.resize((250,250)))#cv2.imshow('fff',render)

image4=Label(self, image=render,borderwidth=15, highlightthickness=5,height=200,
width=200, bg='white')
image4.image = render image4.place(x=750, y=150) contents="Image Loadeded
successfully !!!T = Text(self, height=19, width=25) #T.pack()

T.place(x=1000, y=150) T.insert(END,contents) print(contents) self.Input_img=Input_img

def query2(self, event=None): contents ="Loading Image..."global T,rep2

T = Text(self, height=19, width=25)#T.pack()

T.place(x=1000, y=150)T.insert(END,contents) print(contents)

rep2 = filedialog.askopenfilenames()img = cv2.imread(rep2[0])

cv2.imshow('fff2',img)
img = cv2.resize(img,(256,256))#cv2.imshow('fff1',img) Input_img=img.copy()
print(rep2[0]) cv2.imwrite('fin.png',img) render = Image.open('fin.png')

render = ImageTk.PhotoImage(render.resize((250,250)))#cv2.imshow('fff',render)

image4=Label(self, image=render,borderwidth=15, highlightthickness=5,height=200,
width=200, bg='white')
image4.image = render image4.place(x=500, y=150) contents="Image Loadeded
```

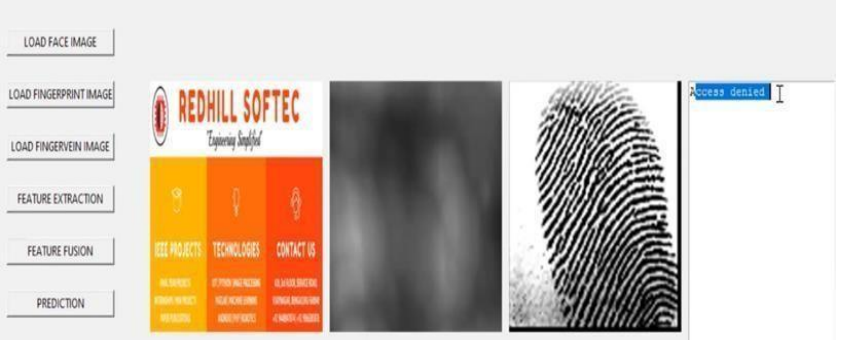
```
successfully !!!T = Text(self, height=19, width=25) #T.pack()
    T.place(x=1000, y=150) T.insert(END,contents) print(contents) self.Input_img=Input_img
    def close_window():Window.destroy()
    def fusion(self):global data
    T = Text(self, height=19, width=25)#T.pack()
    T.place(x=1000, y=150) T.insert(END,"Fusion Completed ..")print(contents)
    def predict(self):
    global data,rep,rep1,rep2 key=np.load('hashing.npy'
        temp_key=np.load('temp_hashing.npy') test_tensors = paths_to_tensor(rep[0])/255
    pred1=model1.predict(test_tensors) print(np.argmax(pred1))
    print(pred1)
        test_tensors = paths_to_tensor(rep1[0])/255pred=model2.predict(test_tensors)
    print(np.argmax(pred))
      test_tensors = paths_to_tensor(rep2[0])/255pred=model3.predict(test_tensors)
  print(np.argmax(pred)) #print(key[0],temp_key)
      if np.max(pred1)>.93 and key[np.argmax(pred1)]==temp_key:contents='Biometric
  accessed successfully ' messagebox.showinfo('Biometric accessed successfully ')
    else:
    contents='Access denied ' messagebox.showinfo('Access denied ')
    T = Text(self, height=19, width=25)#T.pack()
    T.place(x=1000, y=150)T.insert(END,contents) print(contents)
    #data = np.concatenate((face_f,fin_f),axis=0)##print(data)
    root = Tk() root.geometry("1400x800")app = Window(root) root.mainloop()
```

APPENDIX B

SCREENSHOTS







APPENDIX C
STUDENT
CONTRIBUTION

S.NO	ACTIVITY	19K61A05A0	19K61A05B3	20K65A0511	19K65A0565
1	Title Conformation	✓	✓	✓	✓
2	Literature Survey	✓	✓	✓	✓
3	Problem Formulation	✓	✓	✓	✓
4	Requirement Gathering	✓	✓	✓	✓
5	Designing	✓	✓		
6	Implementation	✓		✓	
7	Results and Discussions	✓			
8	Documentation	✓			✓

APPENDIX C

PO, PSO, PEO, AND CO RELEVANCE WITH PROJECT

CO-PO MAPPING SHEET

COURSE OUTCOMES

OUTCOME NO	DESCRIPTION
CO1	Develop problem formation and design skills for engineering and real-world problems.
CO2	Collect and Generate ideas through literature survey on current research areas which help to analyse and present to impart knowledgein different fields.
CO3	Import knowledge on software & hardware to meet industryperspective needs and standards.
CO4	Create interest to carryout research on innovative ideas as a lifelong learning.
CO5	Ability to work with team, and enrich presentation and communication skills.
CO6	Create a platform that makes students employable.

SUMMARY OF CO MAPPING TO PROGRAM OUTCOMES

COs/POs	O1	O2	O3	O4	O5	O6	O7	O8	O9	O10	O11	O12	SO1	SO2
CO1														
CO2														
CO3														
CO4														
CO5														
CO6														
Overall Course														

PROGRAM OUTCOMES (POs)

POs	PROGRAM OUTCOMES	RELEVANCE
PO1	**Engineering Knowledge:** Apply knowledge of mathematics,science, engineering fundamentals, andan engineering specialization to the solution of complex engineering problems.	This project needs a mathematics and Computer Science and Engineering specialization background to perform calculations in the classification task.
PO2	**Problem Analysis:** Identity, formulates, research literature, and analyze complex engineeringproblems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.	For this project ruderous literaturesurvey is conducted to analyze theexisting systems problems.
PO3	**Design/ Development of Solutions:** Design solutions for complex engineeringproblems and design system componentsor processes that meet specified needs with appropriate consideration for publichealth and safety, cultural, societal, and environmental considerations.	Once the formulation of the problem has been completed, the design of the solution relevant to the problem is created to meet the needs of the problem in all aspects.
PO4	**Conduct investigations of complex problems:** Using research-based knowledge and research methods including design of experiments, analysis, and interpretation of data, and synthesis of the information to provide valid conclusions.	Referred to similar kinds of experiments to gain the knowledge of fixing parameters and framing the conclusions.
PO5	**Modern Tool Usage:** Create, select and apply appropriatetechniques, resources, and modernengineering and IT tools includingprediction and modeling to complex engineering activities with an understanding ofthe limitations.	Recent methods like Jupiter notebook have been used to solve the stated problem
PO6	**The Engineer and Society:** Apply To reason informed by thecontextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice.	This problem provides a solution to the people without depletion of any cultural, social, health, safety, and legal issues.

PO7	**Environment and Sustainability:** Understand the impact of professional engineering solutions in societal and environmental contexts and demonstrate knowledge of and need for sustainable development.	This Project doesn't deteriorate anysort of environmental issues.
PO8	**Ethics:** Apply ethical principles and commit to professional ethics and responsibilitie and norms of engineering practice.	This project has been executed b following proper ethics as stated i the engineering practice.
PO9	**Individual and Team Work:** Function effectively as an individual, andas a member or leader in diverse team and multidisciplinary settings.	This project is carried out wit collective teamwork by making th entire project into proper segments.
PO10	**Communication:** Communicate Effectively on comple engineering activities with the engineering community and with societ at large, such as being able to comprehendand write effective reports and desig documentation, make effective presentations and give and receive clear instructions.	Complete information related to the project has been documented for clea understanding.
PO11	**Life-long Learning:** Recognize the need for and have th preparation and ability To engage inindependent and life-long learning in thebroadest context of technological change.	----
PO12	**Project Management and Finance:** Demonstrate knowledge and understanding of engineering and management principles and apply these toone's work, as a member and leader in a team, to manage projects and in multidisciplinary environments.	This work can be enhanced to a larger extent concerning time and other factors.

ROGRAM SPECIFIC OUTCOME (PSOs)

PSOs	Program Specific Outcome	Relevance
PSO1	**Mobile & Web Application Development:** Ability to develop mobile & web applications using J2EE, Android, and J2ME.	-----
PSO2	**Cloud Services:** To deploy virtualized and cloud-based servicesin the organization.	-----

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

PEOs	Program Educational Objectives	RELEVANCE
PEO 1	Graduates will be able to analyze, design, and develop advanced computer applications to provide a solution to real-world problems.	To get the project executed, all the team have done analysis and research-oriented surveys to frame the solution and identify the limitations.
PEO 2	Graduates are well-trained, confident, research-oriented, and industry-ready professionals who are intellectual,ethical, and socially committed.	As implemented the problem with Deep Learning, the latest trend that leads to being well accustomed to the recent technological standards as perindustry requirement.
PEO 3	Graduates will have the technical, communication skills and character that will prepare them for technical and leadership roles.	After completing the project successfully, all the team members can be able to reach a satisfactory level in explaining technical aspects with effective communication.

COURSE OUTCOME (COs)

POs	Course Outcome	POs, PSOs and PEOs Mapped
PO1	Develop problem formation and design skills for engineering and real-world problems	PO1, PO2,PO3, PSO2
PO2	Collect and Generate ideas through literature surveyson current research areas which help to analyze and	PO2, PO3,PO5, PO6
PO3	Import knowledge on software & hardware to meet industryperspective needs and standards.	PO5, PO7,PO8, PO9
PO4	Create interest to research innovative ideas as lifelong learning.	PO11
PO5	Ability to work with a team, and enrich presentationand communication skills.	PO10
PO6	Create a platform that makes students employable.	PO5, PO9, PO11, PO12,PSO2

RELEVANCE TO Pos

CO	PO	PI	Relevance
CO1	PO1	1.2.1	Apply different statistics and numerical techniques to solve the problem.
	PO4	4.4.2	Understand the problem and applied the proper algorithm.
	PO6	6.4.1	This is challenged state to assess societal, safety and legal issues.
	PO7	7.3.1	Identified the risks/impacts in the life-cycle of an product and activity.

	PO8	8.3.1	Identified situations of unethical professional conduct and propose ethical alternatives.
	PO9	9.5.2	This work is carried out by all the team members.
	PO10	10.4.2	Produced the work in well-structured form**.**
	PO12	12.5.2	This work can be enhanced to larger extent with respect to the time and other factors.
CO2	PO1	1.6.1	Uses the engineering fundamentals to complete the work.
	0PO2	2.6.4	Compared and select alternative solution/methods to select the best methods.
	PO6	6.4.1	Interpret legislation, regulations, codes, and standards relevant to your discipline and explain its contribution to the protection of the public.
	PO9	9.5.2	Treat other team members respectfully.
	PO10	10.4.2	Produced the work in well-structured form**.**
	PO1	1.2.1	Applied the knowledge of discrete structures, linear algebra, statistics and numerical techniques to solve problems.

CO3	PO3	3.6.2	Ability to produce a variety of potential design solutions suited to meet functional requirements.
	PO4	4.4.3	Ability to choose appropriate hardware/software tools to conduct the experiment.
	PO5	5.5.1	Identify the strengths and limitations of tools for (i) acquiring information, (ii) modelling and simulating, (iii) monitoring system performance, and (iv) creating engineering designs.
	PO9	9.4.2	Implement the norms of practice (e.g. rules, roles, charters, agendas, etc.) of effective team work, to accomplish a goal.
	PO10	10.4.1	Read, understand and interpret technical and nontechnical information.
CO4	PO1	1.5.1	Apply laws of natural science to an engineering problem.
	PO4	4.6.2	Critically analyse data for trends and correlations, stating possible errors and limitations.
	PO5	5.6.2	Verify the credibility of results from tool use with reference to the accuracy and limitations, and the assumptions inherent in their use.
	PO7	7.4.1	Describe management techniques for sustainable development
	PO8	8.4.2	Examine and apply moral & ethical principles to known case studies
	PO9	9.5.2	Treat other team members respectfully
	PO10	10.5.2	Deliver effective oral presentations to technical and nontechnical audiences
	PO11	11.4.2	Analyze different forms of financial statements to evaluate the financial status of an engineering project

CO5	PO1	1.6.1	Apply engineering fundamentals
	PO5	5.5.2	Demonstrate proficiency in using discipline specific tools.
	PO9	9.5.3	Listen to other members ure in difficult situations
	PO10	10.5.1	Listen to and comprehend information, instructions, and viewpoints of others
	PO12	12.6.2	Analyze sourced technical and popular information for feasibility, viability, sustainability, etc.
CO6	PO1	1.7.1	Apply theory and principles of computer science engineering to solve an engineering problem.
	PO2	2.6.2	Identifies functionalities and computing resources.
	PO5	5.6.1	Discuss limitations and validate tools, techniques and resources
	PO6	6.3.1	Identify and describe various engineering roles; particularly as pertains to protection of the public and public interest at global, regional and local level.
	PO8	8.3.1	Identify situations of unethical professional conduct and propose ethical alternatives.
	PO9	9.5.1	Demonstrate effective communication, problem solving, conflict resolution and leadership skills
	PO10	10.5.1	Listen to and comprehend information, instructions, and viewpoints of others
	PO11	11.6.1	Identify the tasks required to complete an engineering activity, and the resources required to complete the tasks.
	PO12	12.6.1	Source and comprehend technical literature and other credible sources of information.

Object Detection with Audio Feedback to Assist Blind People

CO	PSO	Relevance
CO1	-	-
CO2	-	-
CO3	-	-

CO4	-	-
CO5	-	-
CO6	-	-