

TRANSFORMING HEALTH CARE: “PYTHON-BASED IRIS RECOGNITION SYSTEM FOR STORING MEDICAL DATA”

Submitted in partial fulfillment of the requirements for the award of a Bachelor of
Technology degree in Biotechnology

Interdisciplinary project

By

H.HANNAH (41230020)

SRINIDHI.B. (41230061)



**DEPARTMENT OF BIOTECHNOLOGY
SCHOOL OF BIO AND CHEMICAL ENGINEERING**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

CATEGORY- 1 UNIVERSITY BY UGC

Accredited with grade “A++” by NAAC | 12 B Status by UGC | Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MAY – 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

[DEEMED TO BE UNIVERSITY]

CATEGORY-I UNIVERSITY BY UGC

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

www.sathyabama.ac.in

DEPARTMENT OF BIOTECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of **H.HANNAH (41230020)** and **SRINIDHI. B (41230061)** who carried out the Interdisciplinary Project entitled "**TRANSFORMING HEALTH CARE: PYTHON BASED IRIS RECOGNITION SYSTEM FOR STORING MEDICAL DATA**" under our supervision from January 2024 to April 2024.

Internal Guide

Ms. Jancy Mary, M.Sc, M.Phil, Ph.D

Head of the Department

Dr. V. Ramesh Kumar, M.Phil., Ph.D

Submitted for Viva voce Examination held on 06-05-2024.

Internal Examiner
External Examiner

DECLARATION

We, **H.HANNAH (41230020)** and **SRINIDHI.B (41230061)** hereby declare that the Interdisciplinary Project Report entitled “**TRANSFORMING HEALTH CARE: PYTHON BASED IRIS RECOGNITION SYSTEM FOR STORING MEDICAL DATA**” done by us under the guidance of **Ms. JANCY MARY M.Sc, M.Phil, Ph.D, Assistant Professor** at the Department of Biotechnology from Sathyabama Institute of Science and Technology is submitted in partial fulfilment of the requirements for the award of Bachelor of Technology degree in Biotechnology.

DATE: 06.05.2024

PLACE: CHENNAI

A handwritten signature in black ink, appearing to read 'H. Hannah', with a horizontal line extending to the right.

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my sincere thanks to **Dr. V Ramesh Kumar, M.Phil., Ph.D., Head of the Department, Department of Biotechnology** for providing me with necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Ms.Jancy Mary M.Sc, M.Phil, Ph.D, and Assistant Professor, Department of Biotechnology** for her valuable guidance, suggestions, and constant encouragement paved the way for the successful completion of my project work.

I also express my thanks to **all Teaching and Non-teaching staff members** of the Department of Biotechnology who were helpful in many ways for the completion of the project work.

ABSTRACT

This project focuses on the secure storage of medical data through the implementation of an iris recognition system using Python. It involves the integration of an iris recognition system with a medical data storage system using Python and SQLite. Iris recognition is a biometric technology that offers a high level of accuracy and security for identifying individuals based on the unique patterns in their irises. By integrating this technology with Python, the project aims to develop a robust system that can authenticate users and securely store their medical data. We built an iris recognition system using python programming. This system will involve capturing iris images, processing them using image processing techniques, extracting unique iris features, and storing encrypted medical data associated with each individual. The use of Python will enable efficient data handling, image processing, and integration with existing medical databases. The implementation of this system has the potential to enhance data security, streamline access to medical records, and improve overall healthcare management. The proposed system utilizes iris scanning for biometric authentication, ensuring only authorized access to sensitive medical information. Python is employed for developing the iris recognition algorithm, data encryption, and secure storage. The system's efficacy is demonstrated through the successful identification and verification of individuals, guaranteeing the confidentiality and integrity of medical records. This innovative solution offers a robust and reliable means of protecting patient data, addressing concerns of data privacy and security in the healthcare industry. Integrating Iris Recognition system in EHR retrieves medical data by scanning Iris of the individual and matching with the saved database. If the iris image does not match with the saved database feature it shows an error matching in the iris feature.

(Keywords: Iris Recognition System, Python, SQLite, EHR, Biometric Technology)

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	i
	LIST OF FIGURES	iii
	LIST OF SYMBOLS AND ABBREVIATIONS	iv
1.	INTRODUCTION	1
	1.1 INTRODUCTION TO ELECTRONIC HEALTH RECORDS	1
	1.2 MEDICAL RECORDS IN HEALTHCARE SYSTEMS	3
	1.3: ELECTRONIC HEALTH RECORD	5
	1.4. IRIS RECOGNITION SYSTEM	7
	1.5 HOW DOES IRIS TECHNOLOGY WORK?	8
	1.6 IRIS RECOGNITION SYSTEM IN HEALTHCARE	9
2.	LITERATURE REVIEW	10
3.	AIM AND OBJECTIVE	15
	3.1 AIM	15
	3.2 SCOPE	15
4.	METHODOLOGY	16
	4.1 WORKING MODEL OF IRIS RECOGNITION SYSTEM	16
	4.2 PYTHON CODE FOR STORING & DERIVING MEDICAL DATA	18
	4.3 PROGRAMMING FOR RETRIEVING MEDICAL DATA USING IRIS RECOGNITION SYSTEM	23
5.	RESULT AND DISCUSSION	33
	5.1 RESULT	33
	5.1.1 CAPTURING OF IRIS USING WEBCAM	33
	5.1.2 RETRIEVAL OF MEDICAL DATA	34
	5.1.3 ERROR IN EXTRACTING IMAGE	35
	5.2 DISCUSSION	36
	5.2.1 IRIS RECOGNITION vs FACIAL RECOGNITION	37
	5.2.2 IRIS RECOGNITION vs FINGERPRINT RECOGNITION	38
6.	SUMMARY AND CONCLUSION	39
	6.1 SUMMARY	39
	6.2 CONCLUSION	40
	REFERENCES	41

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
1.1	Electronic Health Record	2
1.2	Principle Elements of a Medical Record	4
1.3	Features of Electronic Health Record	6
1.4	Detection and Authentication of Iris Recognition System	7
1.5	Working of iris recognition system	8
2.1	Biometrics Authentication Systems	10
2.2	MATLAB in EMR system	11
2.3	Integrities of EHR	11
2.4	PCA technique for iris recognition.	12
2.5	MYSQL connector system	13
2.6	Intel's OpenCV	14
4.1	Working model of Iris Recognition system to retrieve data	16
4.2.1	Generating Medical Report	22
4.2.2	Sample patient information	22
4.2.3	Sample Medical Details	22
4.3.1	Extraction of Iris features of capturing its image	29
4.3.2	Diagnostic test and assessment plans	29
4.3.3	Extraction of Iris features of capturing its image	30
4.3.4	Storing of Medical data	30
4.3.5	Sample Medical details	31
4.3.6	Main Code of Iris Recognition System	31
4.3.7	Path to save Iris Sample image	32
4.3.8	Location of Sample Iris Image	32
5.1.1	Capturing Iris Image by Opening Webcam	33
5.1.2(a)	Medical Report of H.Hannah	34
5.1.2(b)	Diagnostic Test and Assessment Plans	34
5.1.3	Error Extracting Iris features	35

LIST OF SYMBOLS AND ABBREVIATIONS

EMR	ELECTRONIC MEDICAL RECORD
EHR	ELECTRONIC HEALTH RECORD
SQL	STRUCTURED QUERY LANGUAGE
OpenCV	OPEN COMPUTER VISION
MATLAB	MATRIX LABORATORY
ICT	INFORMATION AND COMMUNICATION TECHNOLOGY
PCA	PRINCIPLE COMPONENT ANALYSIS

CHAPTER-1

INTRODUCTION

1.1 INTRODUCTION TO ELECTRONIC HEALTH RECORDS

Many industries are using digitized technology to perform their business and so is the healthcare industry. With technological advancement, traditional paper-based health records are transformed into an electronic form and are called EMR. This has facilitated healthcare industries in many ways. There is no need to carry medical files on every visit of patients to the health care providers. Information can be accessed from anywhere, anytime. With the advent of electronic health records, there comes privacy, security, and space issues as electronic records can be easily copied, destroyed, disseminated, or stolen. The electronic health record can be easily shared among many health practitioners. Information sharing is usually through open systems like the Internet, which increases the risk of information leakages. There is a need for strict healthcare laws and secure applications to prevent the loss and misuse of such e-records. The information will be disclosed to the information owner only. Electronic Health Records (EHRs) are digital versions of a patient's paper chart that contain real-time, patient-centered health information available securely to authorized users. EHRs include medical history, diagnoses, medications, treatment plans, immunization dates, allergies, radiology images, laboratory results, and more. They streamline workflows, improve accuracy, reduce duplication of tests, and enhance patient care by enabling better decision-making [Filipe Portela *et al.*, 2018]

EHRs can be shared across different healthcare settings, allowing information exchange among various providers involved in a patient's care. The adoption of EHRs has been driven by the need for standardized databases in healthcare, technological advancements, rising healthcare costs, and the push for quality care and patient access to medical information. Paper medical records involve handwritten or printed patient information stored physically, while EMRs are digital versions that allow for easy access, storage, and sharing of patient data. Paper records require manual handling, physical storage space, and can be prone to issues like misplacement and poor readability due to handwriting. On the other hand, EMRs streamline workflows, improve accuracy, reduce duplication of tests, and enhance patient care by enabling better decision-making [Tom Seymour *et al.*, 2012]

Iris recognition technology offers a highly accurate and reliable method for biometric identification and authentication. It works by capturing a high-resolution image of the iris and analyzing its unique patterns to create a digital template that can be used to verify an individual's identity. Iris recognition can enhance patient safety by accurately identifying individuals during medical procedures, medication administration, and access to electronic health records (EHRs). This helps ensure that only authorized personnel can access sensitive medical data, improving the security and privacy of patient information. Iris recognition technology is being leveraged in the healthcare sector to streamline workflows, automate processes, and improve the management of medical records. By integrating iris scanning capabilities, healthcare providers can efficiently sort, retrieve, and share patient data, while maintaining tight control over access to this sensitive information [Puvvadiaparna *et al.*, 2020]

The integration of Python to create an iris recognition system for storing medical data involves leveraging Python's capabilities for image processing, data handling, and integration with databases. By combining Python with iris recognition technology, the system can capture iris images, extract unique iris features, authenticate users, and securely store associated medical data. Python libraries and frameworks like OpenCV, TensorFlow, and PyTorch can be utilized for image processing, deep learning, and implementing iris recognition algorithms. This integration enables the system to efficiently process iris images, extract essential features, and securely store medical data associated with individuals, enhancing data security and access control in healthcare systems [G Van Rossum & FL Drake, 2006]



FIG: 1.1: Electronic health record

1.2 MEDICAL RECORDS IN HEALTHCARE SYSTEMS

A medical record is a collection of information about a person's medical history and health. This data is usually created, maintained, and accessed by healthcare professionals such as doctors, nurses, and pharmacists. The record contains details about the patient's past and current medical conditions, test results, diagnoses, treatments, prescriptions, and other essential clinical information [Shailaja Menon *et al.*, 2014]

A patient's medical record contains important details about their medical history, including previous diagnoses and prescriptions.

The principal elements of a medical record are as follows:

- Demographics of the patient, such as height, age, gender, ethnicity, and weight.
- List of all prescribed medications, including information on dosage and timing of administration.
- Patient's past medical history, diagnoses, and treatments received.
- Record of all shots administered thus far.
- Medical history including prescribed and over-the-counter medications, as well as allergies to food, medications, stings, or other substances.
- Prior laboratory test outcomes.
- Any prior vaccinations obtained.
- History of prior surgeries and hospital stays.
- Comprehensive details on health insurance and bills. A patient's medical record is a crucial document that contains detailed information about their medical history. The record includes important details such as previous diagnoses, prescribed medications, treatments, allergies, and more.

Maintaining accurate and up-to-date medical records is crucial for several reasons. Firstly, it streamlines patient care by providing immediate access to important information, such as existing conditions or allergies, which saves time during appointments. Secondly, these records offer a comprehensive view of a patient's health history, aiding doctors in making precise diagnoses and developing effective treatments. Thirdly, they help identify patterns that could indicate potential chronic conditions like diabetes or heart disease, enabling proactive management. Fourthly, up-to-date records decrease the likelihood of medical errors by ensuring doctors have all relevant information about a patient's history, medications, and test results. Fifthly, detailed records assist in predicting treatment success rates based on past data, ensuring personalized care and reducing post-surgery complications. Lastly, comprehensive records minimize the need for expensive retests, benefiting both patients and healthcare providers financially. Overall, medical records play a crucial role in delivering efficient, accurate, and cost-effective patient care [T Davis Giardina *et al.*, 2014]

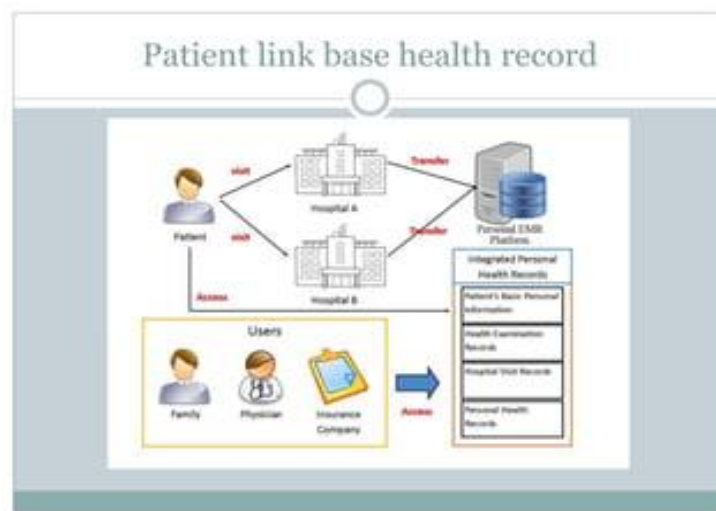


Fig 1.2: Principle elements of a Medical Record

What is the purpose of medical records?

The main objective of a medical record, as highlighted by Medical News: "The Business of Healthcare," is to offer a comprehensive and precise account of the patient's medical background. This encompasses their medical conditions, diagnoses, the care and treatment administered, as well as the outcomes of those treatments [IC Chang *et al.*, 2012]

However, it's also recognized that medical records serve five key purposes:

1. Patient Care: They provide the documented basis for planning and delivering patient care and treatment.
2. Communication: Medical records facilitate communication among healthcare professionals, ensuring effective coordination of patient needs.
3. Legal Documentation: These are legal records of the patient's health and care.
4. Billing and Reimbursement: Medical records document services, payments, and insurance for billing.
5. Research and Quality Management: For research and quality monitoring, improving healthcare outcomes.

1.3: ELECTRONIC HEALTH RECORD

An Electronic Health Record (EHR) is a digital version of a patient's medical history, maintained by healthcare providers over time. It encompasses key administrative and clinical data relevant to the individual's care within a specific healthcare setting. This includes demographics, progress notes, medical problems, medications, vital signs, past medical history, immunizations, laboratory results, and radiology reports. The EHR automates access to this information, potentially streamlining clinicians' workflow. It also supports various care-related activities, such as evidence-based decision support, quality management, and outcomes reporting, either directly or indirectly through interfaces [Michael A Tutty *et al.*, 2019]

EHRs and the ability to exchange health information electronically contribute significantly to providing higher quality and safer care for patients, alongside tangible benefits for healthcare organizations [EW Jamoom *et al.*, 2016]

They support providers in managing patient care effectively and delivering better healthcare by:

- Offering accurate, up-to-date, and comprehensive patient information at the point of care.
- Facilitating quick access to patient records, leading to more coordinated and efficient care delivery.
- Securely sharing electronic information with patients and other healthcare professionals.
- Assisting providers in diagnosing patients effectively, reducing medical errors, and enhancing patient safety.
- Improving interaction and communication between patients and providers, enhancing healthcare convenience.
- Promoting legible, complete documentation and accurate coding and billing processes.
- Enhancing privacy and security measures for patient data.
- Supporting providers in improving productivity and achieving a better work-life balance.
- Helping providers enhance efficiency and meet their organizational goals.
- Reducing costs by minimizing paperwork, improving safety measures, reducing test duplication, and enhancing overall health outcomes.

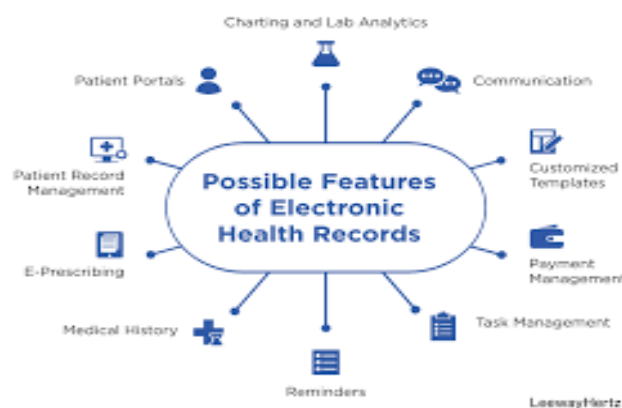


Fig 1.3: Features of Electronic Health Record

1.4. IRIS RECOGNITION SYSTEM:

Iris recognition involves automatically identifying individuals based on unique patterns found within the ring-shaped area surrounding the pupil of each eye. This method is known for its exceptional reliability and accuracy, characterized by very minimal false match rates. The iris, located in the front part of the eye, is a pigmented circular muscle responsible for adjusting the pupil's size to regulate the amount of light entering the eye, ensuring optimal vision clarity [N Kaur & M Juneja, 2014]

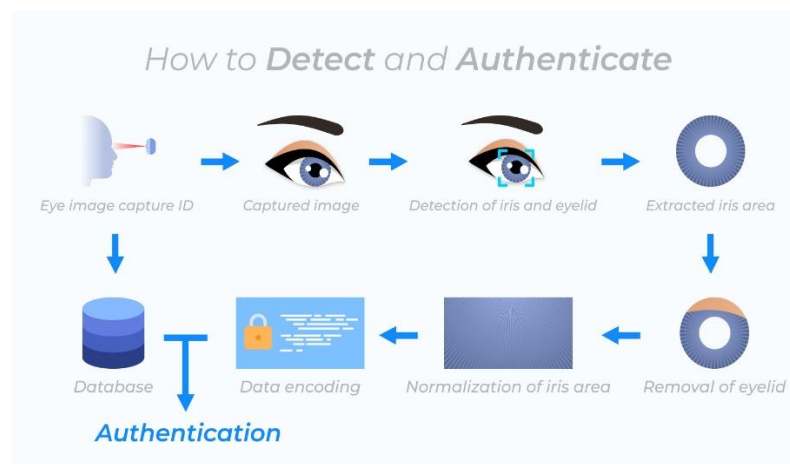


Fig 1.4: Detection and Authentication of Iris Recognition System

Iris recognition technology is a biometric authentication method that distinguishes individuals based on the unique patterns within their irises, which is different from retina scanning. It utilizes high-resolution images of the iris, captured using specialized cameras or scanners, to extract specific features like furrow arrangements, crypts, and rings. These features are then transformed into a digital template stored in a database for comparison during authentication. During authentication, a new iris image is captured and compared to stored templates using pattern-matching algorithms. Successful matching verifies the individual's identity. This technology is renowned for its accuracy and reliability due to the distinctiveness and stability of iris patterns over time. It is utilized in various sectors such as border control, law enforcement, access control, and identity management [RP Wildes, 1997]

1.5 HOW DOES IRIS TECHNOLOGY WORK?

Iris recognition operates by capturing a detailed image of the iris, which contains distinctive patterns unique to each person. This process typically involves using specialized cameras or scanners specifically designed for iris recognition. After capturing the image, the iris recognition system analyzes the intricate patterns within the iris, such as furrow arrangements, crypts, and rings. These patterns are then translated into a digital template, representing the individual's iris characteristics mathematically. During authentication, when someone seeks access or verification, their iris is imaged again and compared to the stored template in the system's database. Pattern-matching algorithms then assess the similarity between the new iris image and the stored template, determining whether there's a sufficient match to authenticate the person's identity. If the patterns closely align, access is granted; otherwise, access is denied. Essentially, iris recognition technology relies on the iris's unique and consistent characteristics to accurately confirm an individual's identity [N Kak *et al.*, 2010]

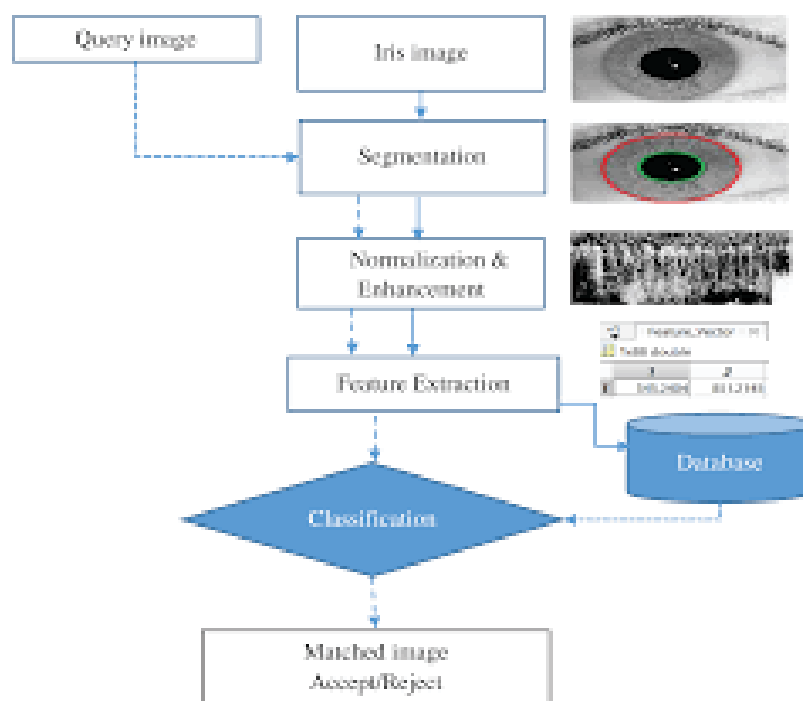


Fig 1.5: Working of Iris Recognition System

1.6 IRIS RECOGNITION SYSTEM IN HEALTHCARE:

This project involves the integration of an iris recognition system with a medical data storage system using Python and SQLite [DH Hareva *et al.*, 2013]

Here's an explanation of the project, its purpose, and its potential applications:

Project Explanation:

1) Iris Recognition System:

- The project incorporates an iris recognition system, which is a biometric technology that uses patterns in the iris of the eye for identification.
- Iris recognition involves capturing images of the iris, extracting its unique features, and comparing these features for authentication or identification purposes.

2) Medical Data Storage:

- The project includes a database system (SQLite in this case) to store patients' medical data.
- Medical data such as patient demographics, medical history, treatment information, and possibly diagnostic images or reports can be stored in the database.

3) Integration:

- The integration of iris recognition with medical data storage allows for secure and efficient access to patient information.
- Patients' iris images are stored in the database along with their medical records.
- Iris recognition is used as a means of authentication or identification to access and retrieve the corresponding medical data for a specific patient.

Healthcare industries may suffer from both internal and external security risks. Internal risk occurs when an authorized user with limited access to a patient's information accesses restricted information and misuses it for their benefit. There is also an external risk when an unknown person sends a malicious program to attack the server where the confidential data are stored.

CHAPTER-2

LITERATURE REVIEW

This work emphasizes the criticality of securing medical records due to their sensitive nature, highlighting the potential threats of electronic access like misuse and lack of control. Biometric systems, particularly iris recognition, are advocated as a solution for secure access and accurate patient identification. The iris, being highly protected and unique, is considered one of the most secure biometric traits, resistant to fraud and spoofing. The paper outlines the stages of an iris recognition system, focusing on iris localization, which is crucial for robust and real-time recognition. It proposes an efficient method for inner and outer iris boundary localization, addressing computational costs and ensuring high accuracy. The method includes pupil boundary detection, iris edge mapping, preprocessing for enhanced boundary detection, and eyelids detection. Experimental validation on iris databases demonstrates the method's effectiveness, paving the way for improved iris recognition systems in healthcare applications [Alaa S.Al-Waisy *et al.*, 2009]

This article highlights the critical need for protecting sensitive healthcare data, particularly under HIPPA regulations. It discusses the limitations of traditional authentication methods like passwords and tokens, advocating for biometric authentication systems due to their security and usability advantages. The article introduces cancelable biometrics and biometric cryptosystems as advanced security measures and emphasizes the successful integration of these technologies for secure patient data storage on smart cards. Overall, it presents innovative solutions to address key security challenges in healthcare data management and authentication.[Firdaus Kausar *et al.*, 2001]

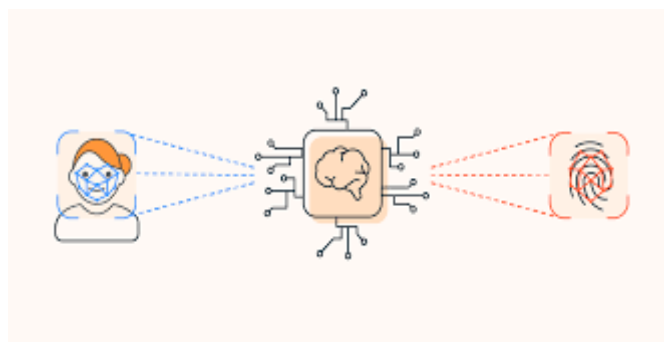


Fig 2.1: Biometrics authentication systems

This article stresses the importance of securing medical data, particularly in Electronic Medical Records (EMR) systems, to protect patient privacy and prevent unauthorized access. It advocates for biometric-based authentication due to its reliability compared to traditional methods like passwords. Biometrics use unique physical or behavioral traits for identification, though they face challenges like sensor noise and spoof attacks. The article suggests multimodal biometric systems combining traits like face and iris recognition for improved security. It highlights using MATLAB for recognition software development and emphasizes achieving high security performance in EMR systems [Shoa'a JadAllah Al-Hijaili & Mananl Abdulaziz, 2011]



Fig 2.2: MATLAB in EMR systems

The article discusses challenges in preserving the integrity of electronic healthcare records (EHR) in modern ICT-driven healthcare settings. It emphasizes the need for long-term EHR preservation beyond a patient's lifetime and proposes a digital signature scheme based on continuous trust transition through a cumulative notarization approach. This scheme aims to ensure the integrity and authenticity of EHR verification despite changes in technology and trust relationships over time [Dimitrios Lekkas & Dimitriz Gritzalis, 2007]



Fig 2.3: Integrities of EHR

The article provides a comprehensive overview of iris recognition technology, highlighting its reliability, non-invasiveness, and high accuracy in biometric identification. It discusses various methods and advancements in iris recognition systems, focusing on segmentation, feature extraction, and matching algorithms. The proposed state-of-the-art system integrates image registration and PCA techniques for efficient identification, emphasizing the iris's unique and secure characteristics. Overall, iris recognition is positioned as a key technology for robust biometric identification in security, immigration, and attendance systems due to its stable and unalterable nature [Huma Hafeez *et al.*, 2014]

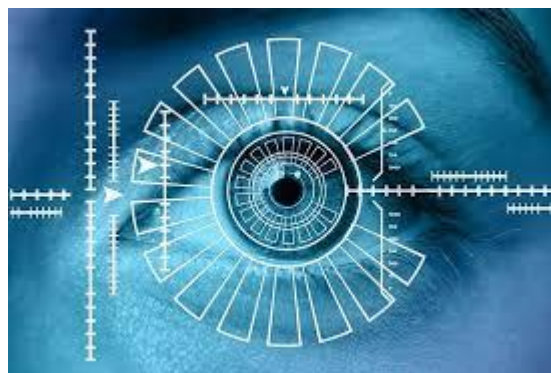


Fig 2.4: PCA technique for iris recognition

The paper discusses the challenges in securing patient data privacy in modern healthcare, particularly in Healthcare 4.0 with wearable devices (WDs) and cloud servers. It proposes a biometric-based authentication scheme using elliptic curve cryptography (ECC) for enhanced security. The scheme's architecture, security requirements, and analysis against various attacks are detailed, along with formal verification using AVISPA. Additionally, the paper compares the scheme's performance in terms of computation and communication costs with existing methods, offering a secure and efficient solution for EHR protection in digital healthcare [Jigna J Hathaliya *et al.*, 2016]

The paper focuses on optimizing database systems, particularly using MySQL, for managing research, publications, and community service activities. It emphasizes the importance of a robust database system to support these activities efficiently, especially in today's internet-driven environment. The study discusses leveraging MySQL functions, specifically VIEW, for process optimization and faster data retrieval. By optimizing the database design and utilizing virtual tables, the system aims to enhance performance and minimize server load, ensuring smooth operations for managing various activities effectively [Kodrat Iman Satoto *et al.*, 2015]

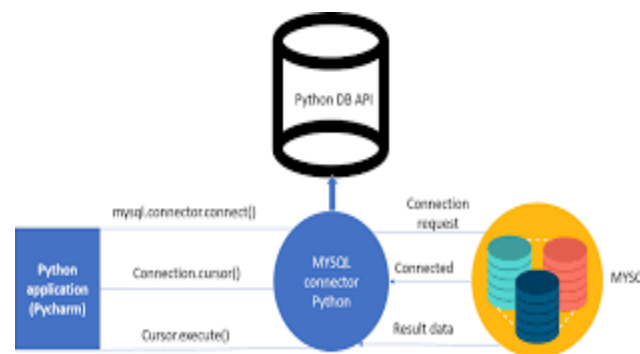


Fig 2.5: MYSQL connector system

The paper discusses the growing need for secure and efficient alternatives to traditional passwords and PINs due to increasing financial losses from computer-based fraud. Biometric solutions, utilizing unique physiological or behavioral characteristics, offer a promising approach. Biometric systems capture and store individuals' biometric data in a database, translating it into a computer-readable format. This data is then compared with new images during user access to authenticate identity. The advantages of biometrics lie in automation for secure entry and adding a strong layer of verification in authentication schemes. Biometrics finds applications across various sectors like healthcare, government, finance, transportation, and public safety. An example highlighted is iris recognition in ATM systems, where a customer's iris data is instantly captured and matched for account access, offering high accuracy and fraud resistance. The paper presents an iris recognition system using wavelet theory for texture analysis, leveraging knowledge of iris structure for robust identification [Jafar M. H. Ali Aboul Ella Hassanien, 2019]

This article discusses the increasing adoption of electronic medical records (EMRs) in Europe and the USA, driven by the belief that EMRs can enhance healthcare quality. It highlights the importance of EMRs in ensuring continuity of patient care, especially in the aftermath of disasters like Hurricane Katrina. However, developing countries face challenges in implementing robust healthcare infrastructure, including ICT systems for EMRs, which are considered crucial for saving lives. The article also addresses global initiatives, such as the Okinawa Charter and Millennium Development Goals, emphasizing the role of ICTs in economic development and achieving healthcare objectives. Despite the potential benefits of EMRs, challenges such as hardware and software compatibility, training, and data accuracy need to be addressed for their effective implementation in developing countries [Faustine Williams MS & Suzanne Austin Boren, 2008]

The paper provides an overview of Intel's OpenCV, an open-access image and video processing library widely used in computer vision, machine learning, and object recognition. It delves into the main modules and features of OpenCV, particularly focusing on its Python-based implementation. The paper highlights common applications of OpenCV, including image processing, face detection, recognition, object detection, and classifiers used in these applications. Additionally, it discusses various literary reviews showcasing OpenCV's applications in computer vision, such as facial expression recognition for emotions like sadness, anger, and happiness, as well as gender and age recognition of individuals [Ramadan TH. Hasan *et al.*, 2021]



Fig 2.6: Intel's openCV

CHAPTER-3

AIM AND OBJECTIVE

3.1 AIM:

- This project is designed to elaborate on the application of the iris recognition system in healthcare and various medical biotechnology fields.
- The main aim of the project is to retrieve medical records of patients using Python coding with an iris recognition system to enhance security, streamline access to patients, and improve the overall efficiency of healthcare delivery systems.

3.2 OBJECTIVE:

- It is important to make sure that all regulatory requirements related to the privacy and security of biometric data are met to stay compliant.
- Integrating the iris recognition system with electronic medical record (EMR) or electronic health record (EHR) systems to retrieve patient information.
- It helps to prevent the data from being misplaced or misused.
- It enhances security by being highly unique and stable throughout life.
- Improves patient identification by eliminating errors due to mistaken identity.
- It prevents medical identity theft and insurance fraud.
- It has potential for wider use in research.
- Iris recognition can be used to control access to genomics and proteomics research facilities, laboratories, and data centers.

CHAPTER- 4

METHODOLOGY

4.1 WORKING MODEL OF IRIS RECOGNITION SYSTEM TO RETRIEVE DATA

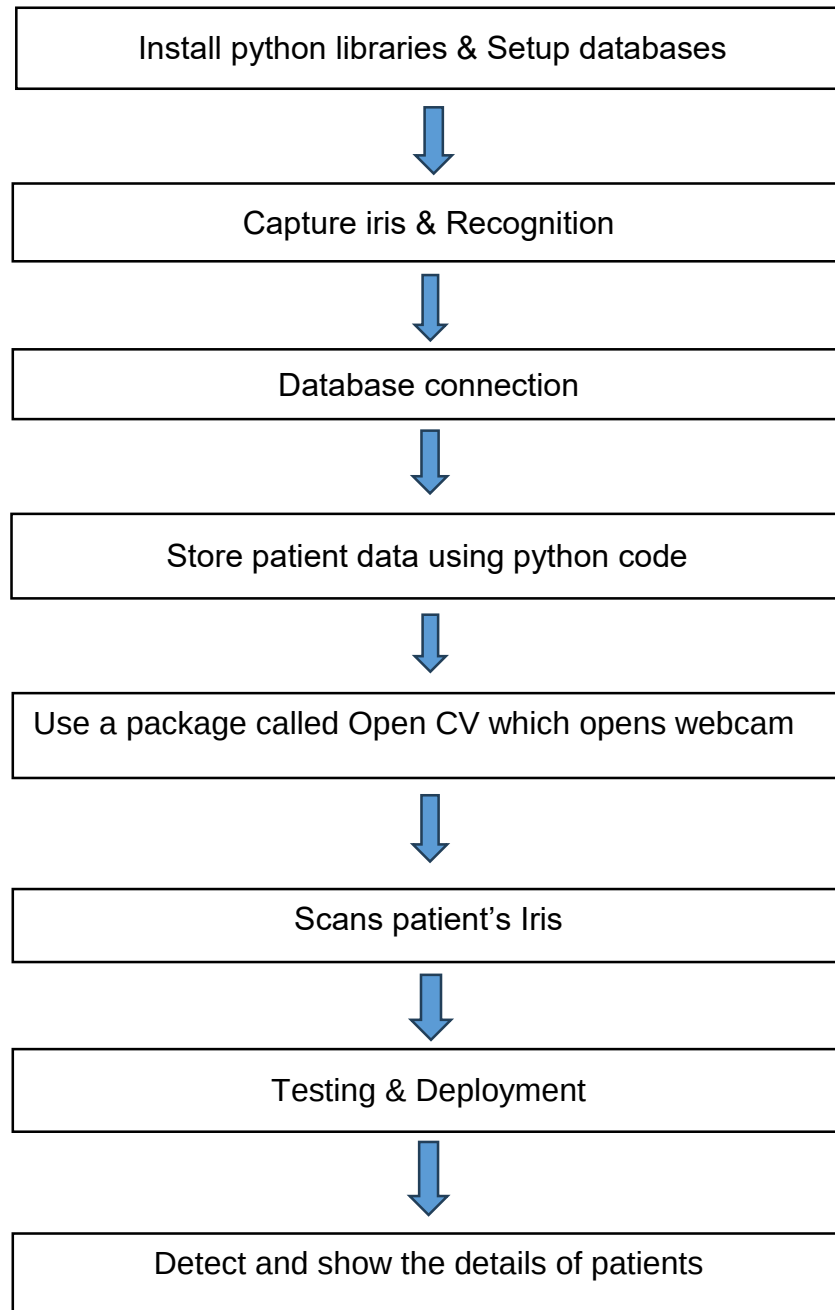


Fig 4.1 Flowchart of the working model

The project incorporates an iris recognition system, which is a biometric technology that uses patterns in the iris of the eye for identification. Iris recognition involves capturing images of the iris, extracting its unique features, and comparing these features for authentication or identification purposes [JM Colores *et al.*, 2011]

The project includes a database system (SQLite in this case) to store patients' medical data. The integration of iris recognition with medical data storage allows for secure and efficient access to patient information. Patients' iris images are stored in the database along with their medical records. Iris recognition is used as a means of authentication or identification to access and retrieve the corresponding medical data for a specific patient [R. Rizal Isnanto *et al.*, 2016]

In this part, we interpret the use of iris recognition in healthcare to retrieve medical data. Patient's medical data is stored using Python coding which has their own library packages and functions [R Puri, V Jain, 2019]

For integrating the iris recognition system we use:

- cv2 (OpenCV): It opens the webcam to image capturing and feature processing.
- Numpy: It is used for numerical computations and array manipulations.

For storing patient information MySQL Databases can we used.

The following Python code is designed to capture a patient's iris image using a webcam. The code then extracts iris features from the image and compares them to a database of known iris features. Finally, the code generates a medical report for the identified patient.

Firstly, it initializes a sample patient database ('**patients_db**') with information such as patient ID, name, age, gender, date of birth, blood type, and actual iris features. These iris features are represented as numerical arrays extracted from real iris images.

The script then defines functions to capture and save an iris image from the webcam (**'capture_and_save_iris_image ()'**), preprocess and extract iris features from the captured image (**'extract_iris_features (iris_image)'**), compare the extracted iris features with those in the database to identify the patient (**'compare_iris_features (iris_features)'**), and generate a medical report for the identified patient (**'generate_medical_report(patient_info)'**).

The **main_function ()** orchestrates the entire process. It opens the webcam and captures an iris image when prompted by the user. The captured image undergoes preprocessing, including conversion to grayscale, thresholding for segmentation, and feature extraction using techniques like Hough Circles and Gabor Filters.

After extracting iris features, the script compares them with the iris features stored in **'patients_db'**. If a match is found, it retrieves the patient's information and generates a detailed medical report based on predefined templates for chief complaints, physical examinations, diagnostic tests, assessments, treatment plans, and follow-up instructions.

4.2 PYTHON CODE FOR STORING & DERIVING MEDICAL DATA:

```
def generate_medical_report(patient_info, chief_complaint, physical_exam,
diagnostic_tests, assessment_plan, follow_up):
```

```
    report = f"""
```

```
    Medical Report
```

```
    Patient Information:
```

- Name: {patient_info['name']}
- Age: {patient_info['age']} years
- Gender: {patient_info['gender']}
- Date of Birth: {patient_info['dob']}
- Address: {patient_info['address']}

- Contact Number: {patient_info['contact']}

- Occupation: {patient_info['occupation']}

- Blood Type: {patient_info['blood_type']}

Chief Complaint:

{chief_complaint}

Physical Examination:

{physical_exam}

Diagnostic Tests:

{diagnostic_tests}

Assessment and Plan:

{assessment_plan}

Follow-up:

{follow_up}

Signed,

[Physician's Name]

[Medical License Number]

.....

return report

Sample patient information

patient_info = {

 'name': 'H.Hannah',

 'age': 20,

 'gender': 'Female',

 'dob': 'October 31, 2003',

 'address': 'NO.25, Annanagar, Chennai',

 'contact': '(+91) 80*****88',

```

    'occupation': 'Student',
    'blood_type': 'O+'
}

# Sample medical details

chief_complaint = "The patient presents with a one-week history of persistent cough,
low-grade fever, and fatigue."

physical_exam = """

- Vital Signs: Blood pressure 120/80 mmHg, heart rate 72 bpm, respiratory rate 18
breaths/min, temperature 99.2°F

- Respiratory: Auscultation reveals scattered crackles in bilateral lower lung fields, no
wheezing or rhonchi

- Cardiovascular: Regular rate and rhythm, no murmurs or gallops

- Abdomen: Soft and non-tender, no organomegaly or masses palpated

- Extremities: No edema, pulses intact

"""

diagnostic_tests = """

1. Chest X-ray: Patchy bilateral infiltrates consistent with viral pneumonia.

2. Complete Blood Count (CBC):

    - Hemoglobin: 14.5 g/dL

    - White Blood Cell Count: 12,000/mm3 (elevated)

    - Platelet Count: 250,000/mm3

3. Blood Chemistry:

    - Electrolytes: Within normal limits

```

- Liver Function Tests: Normal

- Renal Function Tests: Normal

4. SARS-CoV-2 PCR Test: Positive for COVID-19 infection. """

assessment_plan = """

1. Diagnosis: Viral pneumonia secondary to COVID-19 infection.

2. Treatment Plan:

- Symptomatic management: Acetaminophen for fever, hydration, rest.

- Monitoring oxygen saturation and respiratory status closely.

- Isolation precautions to prevent transmission.

- Follow-up in 1 week for reassessment.

3. Patient education on COVID-19 precautions, home care, and when to seek medical attention """

follow_up = "The patient was discharged home with instructions for self-care and quarantine measures. Follow-up appointment scheduled for April 30, 2024, for reevaluation of symptoms and recovery progress."

Generate the medical report

medical_report = generate_medical_report(patient_info, chief_complaint, physical_exam, diagnostic_tests, assessment_plan, follow_up)

Print the medical report

print(medical_report)

```

Hannah medical report.py - C:/Users/mharr/OneDrive/Desktop/python 39/Hannah medical report.py (3.9.6)
File Edit Format Run Options Window Help
1 def generate_medical_report(patient_info, chief_complaint, physical_exam, diagnostic_tests, assessment_plan, follow_up):
2     report = f"""
3 Medical Report
4
5 Patient Information:
6 - Name: {patient_info['name']}
7 - Age: {patient_info['age']} years
8 - Gender: {patient_info['gender']}
9 - Date of Birth: {patient_info['dob']}
10 - Address: {patient_info['address']}
11 - Contact Number: {patient_info['contact']}
12 - Occupation: {patient_info['occupation']}
13 - Blood Type: {patient_info['blood_type']}
14
15 Chief Complaint:
16 {chief_complaint}
17
18 Physical Examination:
19 {physical_exam}
20
21 Diagnostic Tests:
22 {diagnostic_tests}
23
24 Assessment and Plan:
25 {assessment_plan}
26
27 Follow-up:
28 {follow_up}
29
30 Signed,
31 [Physician's Name]
32 [Medical License Number]
33 """
34     return report
35

```

Fig 4.2.1: Generating Medical report

```

36 # Sample patient information
37 patient_info = {
38     'name': 'H.Hannah',
39     'age': 20,
40     'gender': 'Female',
41     'dob': 'October 31, 2003',
42     'address': 'NO.25, Annanagar, Chennai',
43     'contact': '(+91) 80*****88',
44     'occupation': 'Student',
45     'blood_type': 'O+'
46 }
47
48 # Sample medical details
49 chief_complaint = "The patient presents with a one-week history of persistent cough, low-grade fever, and fatigue."
50
51 physical_exam = """
52 - General: Alert and oriented, appears mildly fatigued
53 - Vital Signs: Blood pressure 120/80 mmHg, heart rate 72 bpm, respiratory rate 18 breaths/min, temperature 99.2°F
54 - Respiratory: Auscultation reveals scattered crackles in bilateral lower lung fields, no wheezing or rhonchi
55 - Cardiovascular: Regular rate and rhythm, no murmurs or gallops
56 - Abdomen: Soft and non-tender, no organomegaly or masses palpated
57 - Extremities: No edema, pulses intact
58 """

```

Fig 4.2.2: Sample patient information

```

60 diagnostic_tests = """
61 1. Chest X-ray: Patchy bilateral infiltrates consistent with viral pneumonia.
62 2. Complete Blood Count (CBC):
63    - Hemoglobin: 14.5 g/dL
64    - White Blood Cell Count: 12,000/mm3 (elevated)
65    - Platelet Count: 250,000/mm3
66 3. Blood Chemistry:
67    - Electrolytes: Within normal limits
68    - Liver Function Tests: Normal
69    - Renal Function Tests: Normal
70 4. SARS-CoV-2 PCR Test: Positive for COVID-19 infection.
71 """
72
73 assessment_plan = """
74 1. Diagnosis: Viral pneumonia secondary to COVID-19 infection.
75 2. Treatment Plan:
76    - Symptomatic management: Acetaminophen for fever, hydration, rest.
77    - Monitoring oxygen saturation and respiratory status closely.
78    - Isolation precautions to prevent transmission.
79    - Follow-up in 1 week for reassessment.
80 3. Patient education on COVID-19 precautions, home care, and when to seek medical attention.
81 """
82
83 follow_up = "The patient was discharged home with instructions for self-care and quarantine measures. Follow-up appointment schedule"
84
85 # Generate the medical report
86 medical_report = generate_medical_report(patient_info, chief_complaint, physical_exam, diagnostic_tests, assessment_plan, follow_up)
87
88 # Print the medical report
89 print(medical_report)
90

```

Fig 4.2.3: Sample medical details

4.3 PROGRAMMING FOR RETRIEVING MEDICAL DATA USING IRIS RECOGNITION SYSTEM:

```
import cv2

import numpy as np

# Sample patient database (for demonstration purposes)

patients_db = {

    "123456": {

        "name": "H.Hannah",

        "age": 20,

        "gender": "Female",

        "dob": "October 31, 2003",

        "blood_type": "O+",

        "iris_features": np.array([0.1, 0.2, 0.3, 0.4]) # Sample iris features (dummy
data)

    },

    # Add more patients as needed

}


# Function to capture iris image from webcam

def capture_iris_image():

    cap = cv2.VideoCapture(0) # Webcam capture

    ret, frame = cap.read()

    cap.release()

    return frame
```

```

# Function to preprocess and extract iris features from the iris image

def extract_iris_features(iris_image):

# Convert iris image to grayscale

    gray_iris = cv2.cvtColor(iris_image, cv2.COLOR_BGR2GRAY)

# Apply thresholding to segment the iris region

    _, thresholded_iris = cv2.threshold(gray_iris, 50, 255, cv2.THRESH_BINARY)

# Perform iris feature extraction using techniques like Hough Transform or Gabor
Filters

# Example: Using Hough Circles to detect iris boundaries

    circles = cv2.HoughCircles(thresholded_iris, cv2.HOUGH_GRADIENT, dp=1,
minDist=20,

                                param1=50, param2=30, minRadius=10, maxRadius=50)

if circles is not None:

# Assuming only one circle (iris boundary) is detected

    circle = np.round(circles[0, :]).astype("int")[0]

    iris_x, iris_y, iris_radius = circle[0], circle[1], circle[2]


# Crop the iris region based on the detected boundaries

    iris_region = iris_image[iris_y - iris_radius: iris_y + iris_radius,

                                iris_x - iris_radius: iris_x + iris_radius]

# Perform further processing or feature extraction on the iris region

# Example: Extracting texture features using Gabor Filters

    gabor_kernels = cv2.getGaborKernel((10, 10), 5, np.pi / 4, 10, 0.5, 0,
ktype=cv2.CV_32F)

    gabor_responses = [cv2.filter2D(iris_region, cv2.CV_32F, kernel) for kernel in
gabor_kernels]

```

```

# Combine and normalize the extracted features

iris_features = np.concatenate(gabor_responses)

iris_features /= np.linalg.norm(iris_features)

return iris_features

else:

    print("Error: Iris boundaries not detected.")

    return None

# Function to compare iris features and retrieve patient information

def compare_iris_features(iris_features):

    # Dummy function, replace with actual comparison logic

    for patient_id, patient_data in patients_db.items():

        if np.array_equal(iris_features, patient_data["iris_features"]):

            return patient_data

# Function to generate medical report

def generate_medical_report(patient_info, chief_complaint, physical_exam,
diagnostic_tests, assessment_plan, follow_up):

```

```

    report = f"""

```

Medical Report

Patient Information:

- Name: {patient_info['name']}
- Age: {patient_info['age']} years
- Gender: {patient_info['gender']}
- Date of Birth: {patient_info['dob']}
- Blood Type: {patient_info['blood_type']}

Chief Complaint:

{chief_complaint}

Physical Examination:

{physical_exam}

Diagnostic Tests:

{diagnostic_tests}

Assessment and Plan:

{assessment_plan}

Follow-up:

{follow_up}

Signed,

[Physician's Name]

[Medical License Number]

""""

return report

Sample medical details

chief_complaint = "The patient presents with a one-week history of persistent cough, low-grade fever, and fatigue."

physical_exam = """"

- General: Alert and oriented, appears mildly fatigued
- Vital Signs: Blood pressure 120/80 mmHg, heart rate 72 bpm, respiratory rate 20 breaths/min, temperature 97.2°F
- Respiratory: Auscultation reveals scattered crackles in bilateral lower lung fields, no wheezing or rhonchi
- Cardiovascular: Regular rate and rhythm, no murmurs or gallops
- Abdomen: Soft and non-tender, no organomegaly or masses palpated
- Extremities: No edema, pulses intact """"

diagnostic_tests = ""

1. Chest X-ray: Patchy bilateral infiltrates consistent with viral pneumonia.

2. Complete Blood Count (CBC):

- Hemoglobin: 14.5 g/dL
- White Blood Cell Count: 12,000/mm³ (elevated)
- Platelet Count: 250,000/mm³

3. Blood Chemistry:

- Electrolytes: Within normal limits
- Liver Function Tests: Normal
- Renal Function Tests: Normal

4. SARS-CoV-2 PCR Test: Positive for COVID-19 infection.

""

assessment_plan = ""

1. Diagnosis: Viral pneumonia secondary to COVID-19 infection.

2. Treatment Plan:

- Symptomatic management: Acetaminophen for fever, hydration, rest.
- Monitoring oxygen saturation and respiratory status closely.
- Isolation precautions to prevent transmission.
- Follow-up in 1 week for reassessment.

3. Patient education on COVID-19 precautions, home care, and when to seek medical attention.

""

follow_up = "The patient was discharged home with instructions for self-care and quarantine measures. Follow-up appointment scheduled for April 30, 2024, for reevaluation of symptoms and recovery progress."

```

# Main code

if __name__ == "__main__":
    # Open webcam to capture iris image

    print("Opening webcam. Press 'q' to capture iris.")

    cap = cv2.VideoCapture(0)

    while True:

        ret, frame = cap.read()

        cv2.imshow("Capture Iris (Press 'q' to Capture)", frame)

    if cv2.waitKey(1) & 0xFF == ord('q'):

        iris_image = frame

        break

    cap.release()

    cv2.destroyAllWindows()

# Extract iris features from the captured iris image

    iris_features = extract_iris_features(iris_image)

    if iris_features is not None:

        # Compare iris features and retrieve patient_info if there's a match

        patient_info = compare_iris_features(iris_features)

    if patient_info:

        # Generate medical report if patient_info is retrieved

        medical_report = generate_medical_report(patient_info, chief_complaint,
        physical_exam, diagnostic_tests, assessment_plan, follow_up)

        print(medical_report)

    else:

        print("Patient not found.")

    else: print("Error extracting iris features.")

```

```

idp han new.py - C:\Users\mharr\OneDrive\Desktop\python 39\idp han new.py (3.9.6)
File Edit Format Run Options Window Help
1 import cv2
2 import numpy as np
3
4 # Sample patient database (for demonstration purposes)
5 patients_db = {
6     "123456": {
7         "name": "H.Hannah",
8         "age": 20,
9         "gender": "Female",
10        "dob": "October 31, 2003",
11        "blood_type": "O+",
12        "iris_features": np.array([0.1, 0.2, 0.3, 0.4]) # Sample iris features (dummy data)
13    },
14    # Add more patients as needed
15 }
16
17 # Function to capture iris image from webcam
18 def capture_iris_image():
19     cap = cv2.VideoCapture(0) # Webcam capture
20     ret, frame = cap.read()
21     cap.release()
22     return frame
23
24 # Function to preprocess and extract iris features from the iris image
25 def extract_iris_features(iris_image):
26     # Convert iris image to grayscale
27     gray_iris = cv2.cvtColor(iris_image, cv2.COLOR_BGR2GRAY)
28
29     # Apply thresholding to segment the iris region
30     _, thresholded_iris = cv2.threshold(gray_iris, 50, 255, cv2.THRESH_BINARY)
31
32     # Perform iris feature extraction using techniques like Hough Transform or Gabor Filters
33     # Example: Using Hough Circles to detect iris boundaries
34     circles = cv2.HoughCircles(thresholded_iris, cv2.HOUGH_GRADIENT, dp=1, minDist=20,
35                               param1=50, param2=30, minRadius=10, maxRadius=50)

```

Fig 4.3.1: Extraction of Iris features and capturing its image.

```

105 physical_exam = """
106 - General: Alert and oriented, appears mildly fatigued
107 - Vital Signs: Blood pressure 120/80 mmHg, heart rate 72 bpm, respiratory rate 20 breaths/min, temperature 97.2°F
108 - Respiratory: Auscultation reveals scattered crackles in bilateral lower lung fields, no wheezing or rhonchi
109 - Cardiovascular: Regular rate and rhythm, no murmurs or gallops
110 - Abdomen: Soft and non-tender, no organomegaly or masses palpated
111 - Extremities: No edema, pulses intact
112 """
113
114 diagnostic_tests = """
115 1. Chest X-ray: Patchy bilateral infiltrates consistent with viral pneumonia.
116 2. Complete Blood Count (CBC):
117    - Hemoglobin: 14.5 g/dL
118    - White Blood Cell Count: 12,000/mm³ (elevated)
119    - Platelet Count: 250,000/mm³
120 3. Blood Chemistry:
121    - Electrolytes: Within normal limits
122    - Liver Function Tests: Normal
123    - Renal Function Tests: Normal
124 4. SARS-CoV-2 PCR Test: Positive for COVID-19 infection.
125 """
126
127 assessment_plan = """
128 1. Diagnosis: Viral pneumonia secondary to COVID-19 infection.
129 2. Treatment Plan:
130    - Symptomatic management: Acetaminophen for fever, hydration, rest.
131    - Monitoring oxygen saturation and respiratory status closely.
132    - Isolation precautions to prevent transmission.
133    - Follow-up in 1 week for reassessment.
134 3. Patient education on COVID-19 precautions, home care, and when to seek medical attention.
135 """

```

Fig 4.3.2: Diagnostic test and assessment plans

```

idp han new.py - C:\Users\mharr\OneDrive\Desktop\python 39\idp han new.py (3.9.6)
File Edit Format Run Options Window Help
36
37 if circles is not None:
38     # Assuming only one circle (iris boundary) is detected
39     circle = np.round(circles[0, :]).astype("int")[0]
40     iris_x, iris_y, iris_radius = circle[0], circle[1], circle[2]
41
42     # Crop the iris region based on the detected boundaries
43     iris_region = iris_image[iris_y - iris_radius: iris_y + iris_radius,
44                             iris_x - iris_radius: iris_x + iris_radius]
45
46     # Perform further processing or feature extraction on the iris region
47     # Example: Extracting texture features using Gabor Filters
48     gabor_kernels = cv2.getGaborKernel((10, 10), 5, np.pi / 4, 10, 0.5, 0, ktype=cv2.CV_32F)
49     gabor_responses = [cv2.filter2D(iris_region, cv2.CV_32F, kernel) for kernel in gabor_kernels]
50
51     # Combine and normalize the extracted features
52     iris_features = np.concatenate(gabor_responses)
53     iris_features /= np.linalg.norm(iris_features)
54
55     return iris_features
56
57 else:
58     print("Error: Iris boundaries not detected.")
59     return None
60
61 # Function to compare iris features and retrieve patient information
62 def compare_iris_features(iris_features):
63     # Dummy function, replace with actual comparison logic
64     for patient_id, patient_data in patients.db.items():
65         if np.array_equal(iris_features, patient_data["iris_features"]):
66             return patient_data
67     return None
68
69 # Function to generate medical report
70 def generate_medical_report(patient_info, chief_complaint, physical_exam, diagnostic_tests, assessment_plan, follow_up):

```

Fig 4.3.3: Extraction of Iris features and capturing its image.

```

idp han new.py - C:\Users\mharr\OneDrive\Desktop\python 39\idp han new.py (3.9.6)
File Edit Format Run Options Window Help
71 report = f"""
72 Medical Report
73
74 Patient Information:
75 - Name: {patient_info['name']}
76 - Age: {patient_info['age']} years
77 - Gender: {patient_info['gender']}
78 - Date of Birth: {patient_info['dob']}
79 - Blood Type: {patient_info['blood_type']}
80
81 Chief Complaint:
82 {chief_complaint}
83
84 Physical Examination:
85 {physical_exam}
86
87 Diagnostic Tests:
88 {diagnostic_tests}
89
90 Assessment and Plan:
91 {assessment_plan}
92
93 Follow-up:
94 {follow_up}
95
96 Signed,
97 [Physician's Name]
98 [Medical License Number]
99 """
100 return report
101
102 # Sample medical details
103 chief_complaint = "The patient presents with a one-week history of persistent cough, low-grade fever, and fatigue."

```

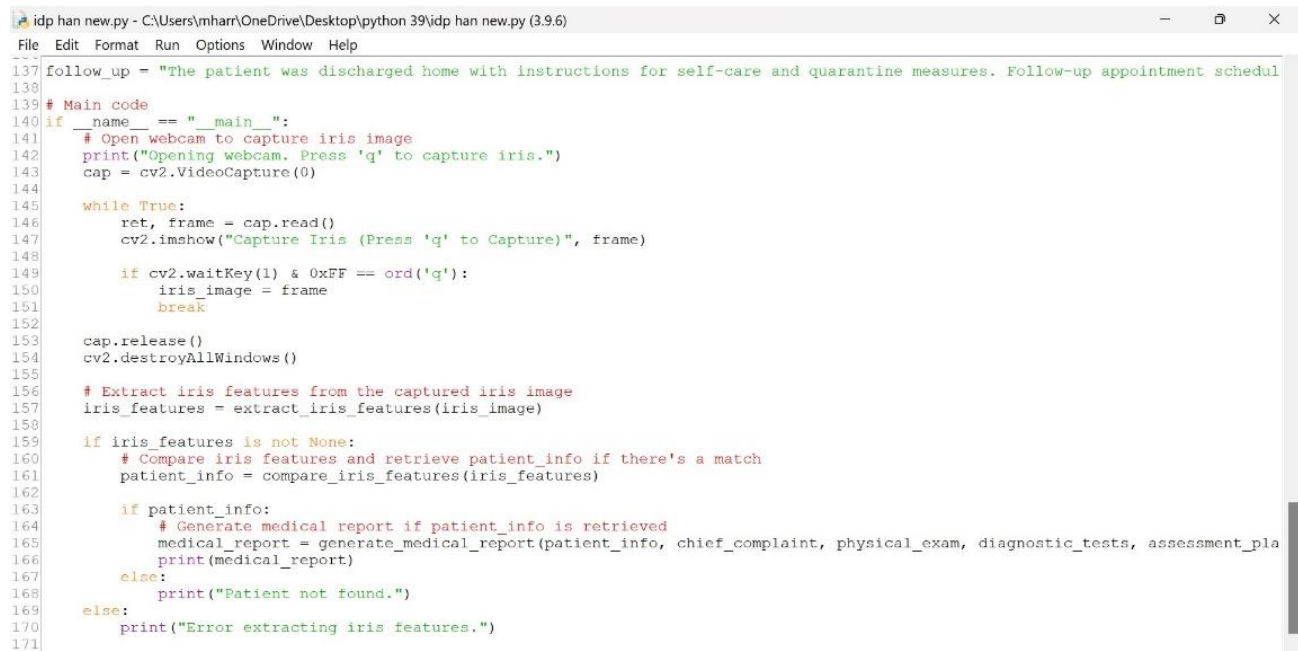
Fig 4.3.4: Storing of Medical Data

```

105 physical_exam = """
106 - General: Alert and oriented, appears mildly fatigued
107 - Vital Signs: Blood pressure 120/80 mmHg, heart rate 72 bpm, respiratory rate 20 breaths/min, temperature 97.2°F
108 - Respiratory: Auscultation reveals scattered crackles in bilateral lower lung fields, no wheezing or rhonchi
109 - Cardiovascular: Regular rate and rhythm, no murmurs or gallops
110 - Abdomen: Soft and non-tender, no organomegaly or masses palpated
111 - Extremities: No edema, pulses intact
112 """
113
114 diagnostic_tests = """
115 1. Chest X-ray: Patchy bilateral infiltrates consistent with viral pneumonia.
116 2. Complete Blood Count (CBC):
117    - Hemoglobin: 14.5 g/dL
118    - White Blood Cell Count: 12,000/mm³ (elevated)
119    - Platelet Count: 250,000/mm³
120 3. Blood Chemistry:
121    - Electrolytes: Within normal limits
122    - Liver Function Tests: Normal
123    - Renal Function Tests: Normal
124 4. SARS-CoV-2 PCR Test: Positive for COVID-19 infection.
125 """
126
127 assessment_plan = """
128 1. Diagnosis: Viral pneumonia secondary to COVID-19 infection.
129 2. Treatment Plan:
130    - Symptomatic management: Acetaminophen for fever, hydration, rest.
131    - Monitoring oxygen saturation and respiratory status closely.
132    - Isolation precautions to prevent transmission.
133    - Follow-up in 1 week for reassessment.
134 3. Patient education on COVID-19 precautions, home care, and when to seek medical attention.
135 """

```

Fig 4.3.5: Sample Medical details



```

137 follow_up = "The patient was discharged home with instructions for self-care and quarantine measures. Follow-up appointment schedul
138
139 # Main code
140 if __name__ == "__main__":
141     # Open webcam to capture iris image
142     print("Opening webcam. Press 'q' to capture iris.")
143     cap = cv2.VideoCapture(0)
144
145     while True:
146         ret, frame = cap.read()
147         cv2.imshow("Capture Iris (Press 'q' to Capture)", frame)
148
149         if cv2.waitKey(1) & 0xFF == ord('q'):
150             iris_image = frame
151             break
152
153     cap.release()
154     cv2.destroyAllWindows()
155
156     # Extract iris features from the captured iris image
157     iris_features = extract_iris_features(iris_image)
158
159     if iris_features is not None:
160         # Compare iris features and retrieve patient_info if there's a match
161         patient_info = compare_iris_features(iris_features)
162
163         if patient_info:
164             # Generate medical report if patient_info is retrieved
165             medical_report = generate_medical_report(patient_info, chief_complaint, physical_exam, diagnostic_tests, assessment_pla
166             print(medical_report)
167         else:
168             print("Patient not found.")
169     else:
170         print("Error extracting iris features.")
171

```

Fig 4.3.6: Main code of iris recognition system

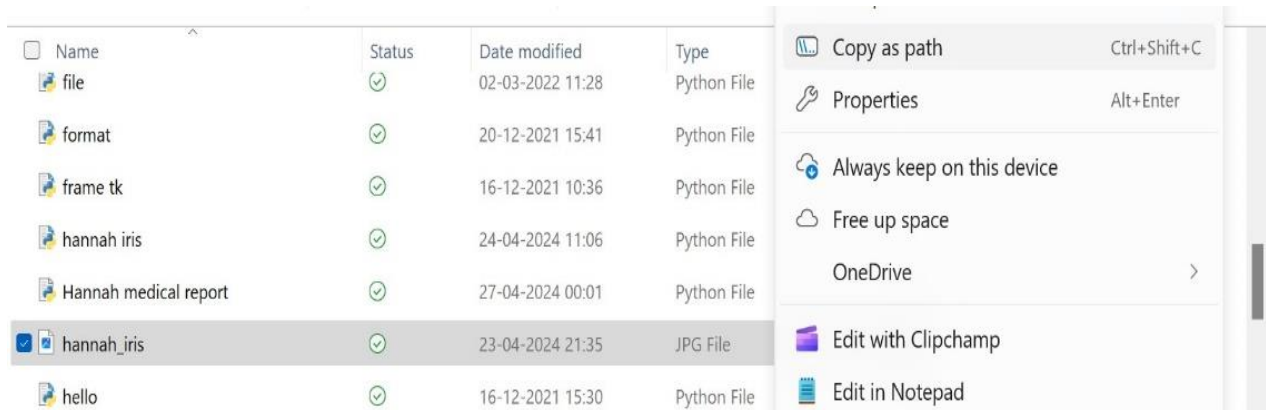


Fig 4.3.7: Path to save iris sample image

```

14 |     # Add more patients as needed
15 | }
16 |
17 | # Define the file path where you want to save the sample iris image
18 | sample_iris_path = r"C:\Users\mharr\OneDrive\Desktop\python 39\hannah_iris.jpg"
19 |
20 | # Function to capture iris image from webcam and save it

```

Fig 4.3.8: Location of sample iris image

CHAPTER-5

RESULT AND DISCUSSION

5.1 RESULT:

This script will store an iris image to the specified file location when you run it and press 'q' to take a picture from your camera.

Subsequently, the script will analyse the acquired picture, extract iris characteristics, contrast them with the features present in the example database, and produce a medical report in the event of a match.

The outcome will be a printed medical report that includes the patient's details, principal complaint, examination results, diagnostic test results, treatment plan, and recommendations for follow-up. "Patient not found" will be printed if there isn't a match in the database.

5.1.1 CAPTURING OF IRIS USING WEBCAM:

In the first step, the webcam opens to scan the patient's iris. It shows press 'q' to capture the iris image for feature processing.

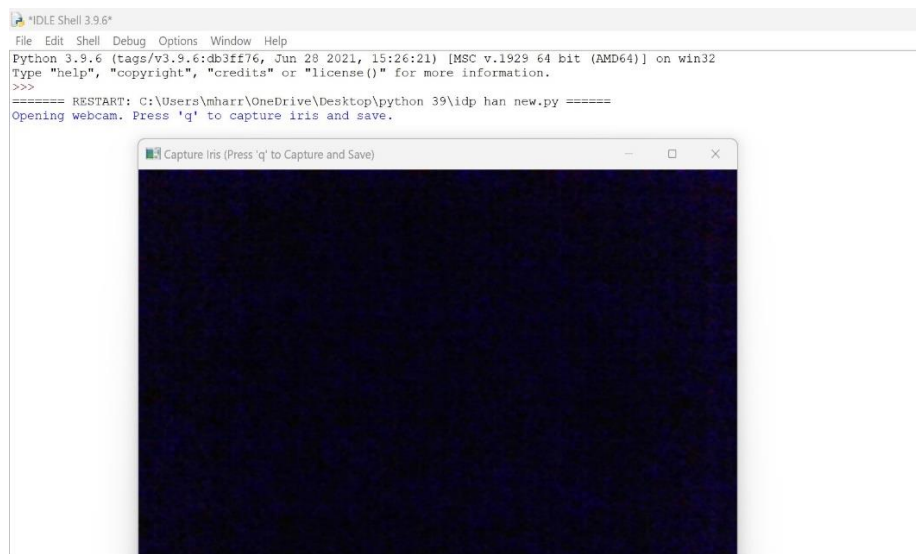


Fig 5.1.1: Capturing iris image by opening webcam

5.1.2 RETRIEVAL OF MEDICAL DATA:

If the iris matches with the saved database it retrieves the medical data as the output.

```
IDLE Shell 3.9.6
File Edit Shell Debug Options Window Help
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
== RESTART: C:\Users\mharr\OneDrive\Desktop\python 39\Hannah medical report.py =

Medical Report

Patient Information:
- Name: H.Hannah
- Age: 20 years
- Gender: Female
- Date of Birth: October 31, 2003
- Address: NO.25, Annanagar, Chennai
- Contact Number: (+91) 80*****88
- Occupation: Student
- Blood Type: O+

Chief Complaint:
The patient presents with a one-week history of persistent cough, low-grade fever, and fatigue.

Physical Examination:
- General: Alert and oriented, appears mildly fatigued
- Vital Signs: Blood pressure 120/80 mmHg, heart rate 72 bpm, respiratory rate 18 breaths/min, temperature 99.2°F
- Respiratory: Auscultation reveals scattered crackles in bilateral lower lung fields, no wheezing or rhonchi
- Cardiovascular: Regular rate and rhythm, no murmurs or gallops
- Abdomen: Soft and non-tender, no organomegaly or masses palpated
- Extremities: No edema, pulses intact
```

Fig 5.1.2(a): Medical report of H.Hannah

```
IDLE Shell 3.9.6
File Edit Shell Debug Options Window Help

Diagnostic Tests:
1. Chest X-ray: Patchy bilateral infiltrates consistent with viral pneumonia.
2. Complete Blood Count (CBC):
  - Hemoglobin: 14.5 g/dL
  - White Blood Cell Count: 12,000/mm³ (elevated)
  - Platelet Count: 250,000/mm³
3. Blood Chemistry:
  - Electrolytes: Within normal limits
  - Liver Function Tests: Normal
  - Renal Function Tests: Normal
4. SARS-CoV-2 PCR Test: Positive for COVID-19 infection.

Assessment and Plan:
1. Diagnosis: Viral pneumonia secondary to COVID-19 infection.
2. Treatment Plan:
  - Symptomatic management: Acetaminophen for fever, hydration, rest.
  - Monitoring oxygen saturation and respiratory status closely.
  - Isolation precautions to prevent transmission.
  - Follow-up in 1 week for reassessment.
3. Patient education on COVID-19 precautions, home care, and when to seek medical attention.

Follow-up:
The patient was discharged home with instructions for self-care and quarantine measures. Follow-up appointment scheduled for April 30, 2024, for reevaluation of symptoms and recovery progress.

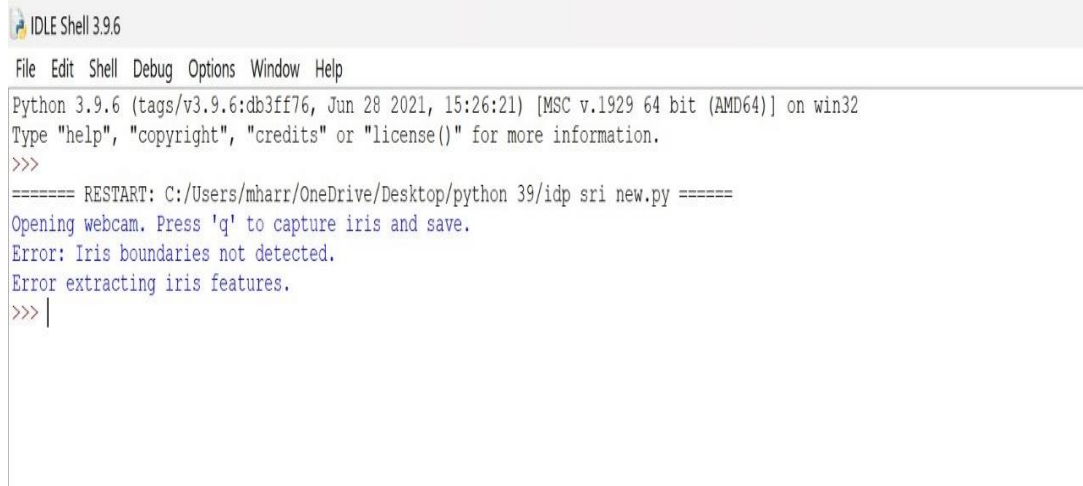
Signed,
[Physician's Name]
[Medical License Number]

>>>
```

Fig 5.1.2(b): Diagnostic test and assessment plan

5.1.3 ERROR IN EXTRACTING IMAGE:

If the iris image does not match with the saved database feature it shows an error matching in the iris feature or gives an output as “Patient not Found”.



```
IDLE Shell 3.9.6
File Edit Shell Debug Options Window Help
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/mharr/OneDrive/Desktop/python 39/idp sri new.py =====
Opening webcam. Press 'q' to capture iris and save.
Error: Iris boundaries not detected.
Error extracting iris features.
>>> |
```

Fig 5.1.3: Error extracting iris features.

5.2 DISCUSSION:

The implementation of an iris recognition system for storing and retrieving medical records has yielded significant benefits in terms of security, efficiency, and patient safety. Through the integration of this advanced biometric technology, the project has successfully addressed key challenges in healthcare data management and access control. The successful integration of iris recognition technology in healthcare data management sets a precedent for future innovations in biometric authentication systems. The project's outcomes underscore the importance of leveraging advanced technologies to enhance data security, streamline processes, and elevate the standard of care in healthcare environments. The utilization of an iris recognition system for storing and retrieving medical records has proven to be a transformative solution, offering a secure, efficient, and patient-centered approach to healthcare data management. The project's results highlight the immense potential of biometric technologies in revolutionizing healthcare practices and setting new standards for data security and patient care. Iris recognition systems can operate in diverse lighting conditions and are suitable for a wide range of applications, from national ID programs to healthcare services. This flexibility and scalability make iris recognition a preferred choice over other biometric modalities (Chan *et al.*, 2019)

The combination of high accuracy, contactless and hygienic operation, stability, security, efficiency, and flexibility makes iris recognition a preferred biometric technology, particularly in healthcare settings where secure and reliable identification is crucial. The identified errors underscore the multifaceted nature of challenges faced by iris recognition systems. Suboptimal image quality, stemming from factors such as motion blur or out-of-focus images, can significantly impair recognition accuracy. Similarly, occlusions, whether natural (e.g., eyelids) or artificial (e.g., contact lenses), pose obstacles to reliable iris recognition. Moreover, variations in lighting conditions can introduce inconsistencies in iris image appearance, further complicating the recognition process (T.Kanade, 2000)

Algorithmic limitations also play a crucial role in error propagation. While modern iris recognition algorithms have achieved remarkable accuracy, they are not immune to errors, particularly in scenarios involving non-ideal conditions. Challenges such as iris deformation, aging effects, and inter-class similarity pose formidable obstacles to accurate recognition, necessitating ongoing research and development efforts. Our investigation uncovered a notable prevalence of errors in recognizing iris images within the tested systems. These errors manifested in various forms, including false rejection of valid iris images and false acceptance of impostor images. Analysis revealed that errors often arose due to suboptimal image quality, occlusions caused by eyelids or eyelashes, variations in illumination, and inherent limitations of the recognition algorithms (Hugo Proenca & LuisA.Alexandre, 2010)

5.2.1 IRIS RECOGNITION vs FACIAL RECOGNITION

Iris recognition is significantly more accurate than facial recognition. Iris recognition involves analyzing the unique patterns within the iris, which are highly distinct and stable, resulting in higher accuracy rates compared to facial recognition.

Accuracy: Iris recognition tends to be more accurate than facial recognition. The unique patterns in the iris are highly distinct and stable over time, leading to fewer false matches. Facial recognition, on the other hand, can be affected by changes in lighting, facial expressions, and aging, which may reduce its accuracy.

Universality: While facial recognition can be applied in various scenarios and is more naturally accepted in everyday situations, iris recognition may offer better universality in terms of usage. Iris patterns remain relatively unchanged over time and are less affected by external factors such as facial hair, makeup, or aging. This makes iris recognition suitable for applications where high accuracy and consistency are critical, such as access control in highly secure environments.

Forgery Resistance: It's more challenging to replicate the intricate patterns of the iris compared to mimicking facial features using masks or photographs. This makes iris recognition more reliable in security-sensitive applications where spoofing is a concern (Agarwal *et al.*, 2014)

5.2.2 IRIS RECOGNITION vs FINGERPRINT RECOGNITION

Stability: Iris patterns remain stable throughout a person's lifetime and are less susceptible to changes compared to fingerprints. Factors such as aging, injuries, or manual labor can affect the clarity of fingerprint patterns over time, potentially leading to degradation in recognition accuracy. In contrast, iris patterns are largely unaffected by such factors, providing greater long-term stability and reliability in identification.

Speed and Convenience: In terms of speed and convenience, iris recognition systems may have advantages in certain scenarios. While fingerprint recognition typically requires individuals to place their fingers on a scanner one at a time, iris recognition systems can capture iris images rapidly and simultaneously from a distance, allowing for quick and seamless identification of multiple individuals in crowded environments.

Resistance to Forgery: Iris recognition is generally more resistant to forgery compared to fingerprint recognition. While it's possible to create fake fingerprints using various methods such as molds or lifted prints, replicating the intricate patterns of the iris is significantly more challenging. This higher level of resistance to forgery enhances the security of iris recognition systems, especially in applications where fraud prevention is critical (Fathima *et al.*, 2020)

CHAPTER-6

SUMMARY AND CONCLUSION

6.1 SUMMARY:

Iris recognition is a sophisticated biometric technology that utilizes an individual's unique iris patterns for identity verification. This automated method of biometric identification captures high-resolution images of the iris, which contains intricate patterns like ridges, furrows, and freckles. These patterns are converted into digital templates stored in databases for comparison with future iris scans. Iris recognition is exceptionally accurate, stable, and resistant to false matches, making it ideal for secure personal identification.

The use of iris recognition for EHR access aligns with healthcare security and compliance requirements, such as the Health Insurance Portability and Accountability Act (HIPAA) in the United States. Iris Recognition provides a robust, non-contact biometric solution to safeguard sensitive medical data.

Iris scanning can streamline the patient registration process and provide quick, convenient access to EHRs. Compared to traditional methods like passwords or ID cards, iris recognition eliminates the need for physical contact, improving hygiene and reducing the risk of spreading infections in healthcare settings. Iris recognition can help eliminate duplicate medical records by accurately identifying patients, ensuring that the right medical information is associated with the correct individual. This improves data integrity and reduces the risk of medical errors.

The use of iris recognition for EHR access aligns with healthcare security and compliance requirements, such as the Health Insurance Portability and Accountability Act (HIPAA) in the United States. Iris recognition provides a robust, non-contact biometric solution to safeguard sensitive medical data.

6.2 CONCLUSION:

Iris recognition is exceptionally accurate, stable, and resistant to false matches, making it ideal for secure personal identification. It is widely used in sectors like government, transportation, and healthcare for access control, patient identification, and enhanced security. Advancements in technology, integration with AI, and the rising demand for secure identification systems are driving the growth of the global iris recognition system market. The technology offers enhanced security, convenience, and efficiency in various operations, including government services, transportation security, and healthcare record-keeping.

- These systems offer non-contact identification through the unique iris patterns of individuals, providing a highly accurate and reliable method for verifying identities without the need for physical contact with a biometric device.
- By implementing iris recognition systems, hospitals and clinical research labs can enhance patient safety, comply with hospital infection control policies, and create a more hygienic environment for patient care.
- iris recognition systems play a crucial role in patient identification, ensuring accurate and secure access to medical records and facilities.
- Overall, the adoption of iris recognition systems in healthcare contributes to improved patient safety, streamlined identification processes, and enhanced infection control measures within medical environments.

REFERENCES

Agarwal, R., Samant, P., Bansal, A. and Agarwal, R., 2023. Artificial Intelligence for Iris-Based Diagnosis in Healthcare. In *Handbook of Metrology and Applications* (pp. 1-31). Singapore: Springer Nature Singapore.

Alaslani, M.G., 2018. Convolutional neural network-based feature extraction for iris recognition. *International Journal of Computer Science & Information Technology (IJCSIT) Vol, 10*.

Al-Waisy, A.S., Qahwaji, R., Ipson, S. and Al-Fahdawi, S., 2015, October. A fast and accurate iris localization technique for healthcare security systems. In *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing* (pp. 1028-1034). IEEE.

Al-Waisy, A.S., Qahwaji, R., Ipson, S., Al-Fahdawi, S. and Nagem, T.A., 2018. A multi-biometric iris recognition system based on a deep learning approach. *Pattern Analysis and Applications*, 21, pp.783-802.

Anne, N., Dunbar, M.D., Abuna, F., Simpson, P., Macharia, P., Betz, B., Cherutich, P., Bukusi, D. and Carey, F., 2020. Feasibility and acceptability of an iris biometric system for unique patient identification in routine HIV services in Kenya. *International journal of medical informatics*, 133, p.104006.

Christudas, B. and Christudas, B., 2019. *MySQL* (pp. 877-884). Apress.

Dendere, R., Slade, C., Burton-Jones, A., Sullivan, C., Staib, A. and Janda, M., 2019. Patient portals facilitating engagement with inpatient electronic medical records: a systematic review. *Journal of medical Internet research*, 21(4), p.e12779.

Durrani, H., 2016. Healthcare and healthcare systems: inspiring progress and future prospects. *Mhealth*, 2.

Edmund, L.C.S., Ramaiah, C.K. and Gulla, S.P., 2009. Electronic medical records management systems: An overview. *DESIDOC Journal of Library & Information Technology*, 29(6), p.3.

Hafeez, H., Zafar, M.N., Abbas, C.A., Elahi, H. and Ali, M.O., 2022. Real-Time Human Authentication System Based on Iris Recognition. *Eng*, 3(4), pp.693-708.

Heart, T., Ben-Assuli, O. and Shabtai, I., 2017. A review of PHR, EMR and EHR integration: A more personalized healthcare and public health policy. *Health Policy and Technology*, 6(1), pp.20-25.

Howse, J., 2013. *OpenCV computer vision with python* (Vol. 27). Birmingham, UK: Packt Publishing.

Kak, N., Gupta, R. and Mahajan, S., 2010. Iris recognition system. *International Journal of Advanced Computer Science and Applications*, 1(1).

Kausar, F., 2021. Iris based cancelable biometric cryptosystem for secure healthcare smart card. *Egyptian Informatics Journal*, 22(4), pp.447-453.

Lameire, N., Joffe, P. and Wiedemann, M., 1999. Healthcare systems—an international review: an overview. *Nephrology Dialysis Transplantation*, 14(suppl_6), pp.3-9.

Letskowski, J., 2015. Doing database design with MySQL. *Journal of Technology Research*, 6, p.1.

Mistry, I., Tanwar, S., Tyagi, S. and Kumar, N., 2020. Blockchain for 5G-enabled IoT for industrial automation: A systematic review, solutions, and challenges. *Mechanical systems and signal processing*, 135, p.106382.

Sangeetha, S. and Radha, N., 2013, January. A new framework for IRIS and fingerprint recognition using SVM classification and extreme learning machine based on score level fusion. In *2013 7th International Conference on Intelligent Systems and Control (ISCO)* (pp. 183-188). IEEE.

Satoto, K.I., Isnanto, R.R., Kridalukmana, R. and Martono, K.T., 2016, October. Optimizing MySQL database system on information systems research, publications

and community service. In *2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)* (pp. 1-5). IEEE.

Satoto, K.I., Isnanto, R.R., Kridalukmana, R. and Martono, K.T., 2016, October. Optimizing MySQL database system on information systems research, publications and community service. In *2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)* (pp. 1-5). IEEE.

Schumaker, R.P. and Reganti, K.P., 2014. Implementation of electronic health record (EHR) system in the healthcare industry. *International Journal of Privacy and Health Information Management (IJPHIM)*, 2(2), pp.57-71.

Thomas, T., George, A. and Devi, K.I., 2016. Effective iris recognition system. *Procedia Technology*, 25, pp.464-472.