**Introduction to Biometric System:**

Identity verification in computer systems is done based on measures like keys, cards, passwords, PIN and so forth. Unfortunately, these may often be forgotten, disclosed or changed. A reliable and accurate identification/verification technique may be designed using biometric technologies, which are further based on the special characteristics of the person such as face, iris, fingerprint, signature and so forth. This technique of identification is preferred over traditional passwords and PIN-based techniques for various reasons:

* The person to be identified is required to be physically present at the time of identification.
* Identification based on biometric techniques obviates the need to remember a password or carry a token.

A biometric system essentially is a pattern recognition system that makes a personal identification by determining the authenticity of a specific physiological or behavioral characteristic possessed by the user. Biometric technologies are thus defined as the “automated methods of identifying or authenticating the identity of a living person based on a physiological or behavioral characteristic.” A biometric system can be either an identification system or a verification (authentication) system; both are defined below.

* **Identification:** *One to Many* —A comparison of an individual’s submitted biometric sample against the entire database of biometric reference templates to determine whether it matches any of the templates.
* **Verification:** *One to One* —A comparison of two sets of biometrics to determine if they are from the same individual.

Biometric authentication requires comparing a registered or enrolled biometric sample (biometric template or identifier) against a newly captured biometric sample (for example, the one captured during a login). This is a three-step process ( *Capture, Process, Enroll* ) followed by a *Verification* or *Identification* .

During *Capture*, raw biometric is captured by a sensing device, such as a fingerprint scanner or video camera; then, distinguishing characteristics are extracted from the raw biometric sample and converted into a processed biometric identifier record (biometric template). Next is enrollment, in which the processed sample (a mathematical representation of the template) is stored/registered in a storage medium for comparison during authentication. In many commercial applications, only the processed biometric sample is stored. The original biometric sample cannot be reconstructed from this identifier.

**BACKGROUND**

Many biometric characteristics may be captured in the first phase of processing. However, automated capturing and automated comparison with previously stored data requires the following properties of biometric characteristics:

* **Universal:** Everyone must have the attribute. The attribute must be one that is seldom lost to accident or disease.
* **Invariance of properties:** They should be constant over a long period of time. The attribute should not be subject to significant differences based on age or either episodic or chronic disease.
* **Measurability:** The properties should be suitable for capture without waiting time and it must be easy to gather the attribute data passively.
* **Singularity:** Each expression of the attribute must be unique to the individual. The characteristics should have sufficient unique properties to distinguish one person from any other. Height, weight, hair and eye color are unique attributes, assuming a particularly precise measure, but do not offer enough points of differentiation to be useful for more than categorizing.
* **Acceptance:** The capturing should be possible in a way acceptable to a large percentage of the population. Excluded are particularly invasive technologies; that is, technologies requiring a part of the human body to be taken or (apparently) impairing the human body.
* **Reducibility:** The captured data should be capable of being reduced to an easy-to-handle file.
* **Reliability and tamper-resistance:** The attribute should be impractical to mask or manipulate. The process should ensure high reliability and reproducibility.
* **Privacy:** The process should not violate the privacy of the person.
* **Comparable:** The attribute should be able to be reduced to a state that makes it digitally comparable to others. The less probabilistic the matching involved, the more authoritative the identification.
* **Inimitable:** The attribute must be irreproducible by other means. The less reproducible the attribute, the more likely it will be authoritative.

Among the various biometric technologies being considered are fingerprint, facial features, hand geometry, voice, iris, retina, vein patterns, palm print, DNA, keystroke dynamics, ear shape, odor, signature and so forth.

**Fingerprint**

Fingerprint biometric is an automated digital version of the old ink-and-paper method used for more than a century for identification, primarily by law enforcement agencies (Maltoni, 2003). The biometric device requires each user to place a finger on a plate for the print to be read. Fingerprint biometrics currently has three main application areas: large-scale Automated Finger Imaging Systems (AFIS), generally used for law enforcement purposes; fraud prevention in entitlement programs; and physical and computer access. A major advantage of finger imaging is the long-time use of fingerprints and its wide acceptance by the public and law enforcement communities as a reliable means of human recognition. Others include the need for physical contact with the optical scanner, possibility of poor-quality images due to residue on the finger such as dirt and body oils (which can build up on the glass plate), as well as eroded fingerprints from scrapes, years of heavy labor or mutilation.

## Performance Measurements

The overall performance of a system can be evaluated in terms of its storag e, speed and accurac y. The size of a template, especially when using smart cards for storage, can be a decisive issue during the selection of a biometric system. Iris scan is often preferred over fingerprinting for this reason. Also, the time required by the system to make an identification decision is important, especially in real-time applications such as ATM transactions.

Accuracy is critical for determining whether the system meets requirements and, in practice, the way the system responds. It is traditionally characterized by two error statistics: False Accept Rate ( *FAR*) (sometimes called False Match Rate), the percentage of impostors accepted; and False Reject Rate ( *FRR*), the percentage of authorized users rejected. These error rates come in pairs: For each false-reject rate there is a corresponding false alarm. In a perfect biometric system, both rates should be zero. Unfortunately, no biometric system today is flawless, so there must be a trade-off between the two rates. Usually, civilian applications try to keep both rates low. The error rate of the system when FAR equals FRR is called the Equal Error Rate , and is used to describe performance of the overall system. Good biometric systems have error rates of less than 1%. This should be compared to error rates in current methods of authentication, such as passwords, photo IDs, handwritten signatures and so forth. Although this is feasible in theory, practical comparison between different biometric systems when based on different technologies is very hard to achieve. The problem with the system is that people’s physical traits change over time, especially with alterations due to accident or aging. Problems can occur because of accident or aging, humidity in the air, dirt and sweat (especially with finger or hand systems) and inconsistent ways of interfacing with the system.

The goal of a technology evaluation is to compare competing algorithms from a single technology. The use of test sets allows the same test to be given to all participants. The goal of scenario testing is to determine the overall system performance in a single prototype or simulated application to determine whether a biometric technology is sufficiently mature to meet performance requirements for a class of applications. The goal of operational testing is to determine the performance of a complete biometric system in a specific application environment with a specific target population, to determine if the system meets the requirements of a specific application.

**Coding**

**public class FingServer {**

**int portno=8078;**

**String server="localhost";**

**void startServer() throws IOException{**

**ServerSocket server = new ServerSocket(portno);**

**Socket client;**

**System.out.println("Server Started");**

**while(true){**

**try{**

**System.out.println("Waiting for client");**

**client = server.accept();**

**System.out.println("Connected to client");**

**branchClient(client);**

**}catch(Exception e){**

**e.printStackTrace();**

**}**

**}**

**}**

**String getHeader( String result){**

**String head ="HTTP/1.1 200 OK\r\n"+**

**"Date: Wed, 20 Dec 2017 06:47:51 GMT\r\n"+**

**"Server: Apache/2.4.27 (Win32) OpenSSL/1.0.2l PHP/7.1.8\r\n"+**

**"Last-Modified: Mon, 27 Nov 2017 09:02:54 GMT\r\n"+**

**"Content-Length:"+(result.length()+2)+" \r\n"+**

**"Keep-Alive: timeout=5, max=100\r\n"+**

**"Connection: Keep-Alive\r\n"+**

**"Content-Type: text/html\r\n\r\n";**

**return head;**

**}**

**void branchClient(Socket client) throws IOException{**

**try{**

**HashMap<String, String> query;**

**System.out.println("Inside Client branch");**

**BufferedReader clientReader = new BufferedReader(new InputStreamReader(client.getInputStream()));**

**PrintWriter out= new PrintWriter(client.getOutputStream());**

**StringBuffer sb = new StringBuffer();**

**String line="";**

**while((line=clientReader.readLine()).length()!=0){**

**System.out.println("Line "+line);**

**sb.append(line);**

**}**

**char buf[] =new char[500];**

**int l =clientReader.read(buf);**

**String result="";**

**String input=new String(buf);**

**System.out.println("data "+input + " l = "+l);**

**query = parseQuery(input);**

**// if(input.contains("register")){**

**// System.out.println("inside ");**

**// result = ""+FingerTest.checkDuplicate();**

**// System.out.println("RES "+result);**

**// }**

**System.out.println("asd "+query);**

**String type=query.get("type").trim();**

**System.out.println("sss "+type+"| "+type.length());**

**if(type.equalsIgnoreCase("register")){**

**System.out.println("inside ");**

**result = ""+FingerTest.checkDuplicate();**

**System.out.println("dup RES "+result);**

**}**

**if(type.equalsIgnoreCase("verify")){**

**String file1,file2;**

**file1=query.get("file1");**

**file2=query.get("file2");**

**file1= URLDecoder.decode(file1, "UTF-8");**

**file2= URLDecoder.decode(file2, "UTF-8");**

**System.out.println("FILE "+file1+" "+file2);**

**result = ""+FingerTest.verifyFP(file1, file2);**

**System.out.println("veridy RES "+result);**

**}**

**System.out.println("Reading client "+sb.toString());**

**//String result ="Hello OK";**

**out.print(getHeader(result));**

**out.print(result);**

**//out.println();**

**out.close();**

**clientReader.close();**

**}catch(Exception e){**

**e.printStackTrace();**

**}**

**}**

**HashMap<String, String> parseQuery(String query){**

**HashMap<String, String> result = new HashMap<>();**

**String ss[]=query.split("&");**

**String kval[];**

**for(int i=0;i<ss.length;i++){**

**kval=ss[i].split("=");**

**result.put(kval[0],kval[1]);**

**}**

**return result;**

**}**

**public static void main(String arg[]){**

**FingServer fs = new FingServer();**

**try {**

**fs.startServer();**

**} catch (IOException ex) {**

**ex.printStackTrace();**

**}**

**}**