Iris Recognition and its Protection Overtone using Cryptographic Hash Function

Gatheejathul Kubra.J*, Rajesh.P**
*PG Student, CSE Department, Mohamed Sathak A.J College of Engineering, Chennai, India
** HOD, CSE Department, Mohamed Sathak A.J College of Engineering, Chennai, India

Abstract--*To overcome the problems faced in security there are many advanced techniques used nowadays. People individually use their finger prints, voice, face reactions, eyes as a password for security purposes .Iris is the part of eye which is unique for every human being. Iris recognition is the secured technique being used for adhar card. Here we use cryptographic hashing function that is used to verify data integrity through the creation of a 128-bit message digest from data input. The objective of 'IRIS Recognition' is primarily to improve the security of biometric recognition technology that uses IRIS. Though it enjoys clear advantage over other methods using finger prints, voice recognition, face recognition etc. in current technology. After the conversion of iris patterns into linear graph, the threshold value is taken which makes it prone to hacking. It is this vulnerability that the proposed technology will address by assigning a random number to the mean value. Thus the proposed technology will be a step forward in enhancing the security of iris based biometric systems.*

Keywords—*Iris Recognition, Edge Detection, Pupil detection, Normalization, Feature extraction*

I. INTRODUCTION

A system for applying pattern recognition techniques to recognize the individuality of a person based on their iris is proposed. Also deliberated is a transform of the iris image from two to one dimensional space and overcoming limited data with the generation of mock images. A modern emphasis on security has resulted in increased research attention being accessible to the field of single identification based on "biometrics".

A biometric feature is an intrinsic physical or behavioral trait that is single among individuals. In accumulation to these, the human iris can also be pondered a valid biometric feature for individual identification. The iris is the colored ring on the human eye in the middle of the pupil and the white sclera. Each human iris has a unique "Iris Code" of elusive features that varies greatly from person to person. Iris features remain endless over an individual's lifetime and are not focus to changes produced by the possessions of elderly as other biometric features may be. For these reasons, the human iris is an idyllic feature for extremely exact and well-organized identification systems. The individuality of iris texture lies in the fact that the processes engendering those textures are wholly frantic but stable. Hence in order to use the iris as a biometric, the feature extraction should be able to capture and encode this randomness present in the iris texture. Founded on an wide-ranging literature survey, we classify iris recognition systems into three groupings depending on the technique by which the features from the texture are extracted for equivalent purposes. These three categories are (a) appearance based, (b) texture based and, (c) feature based extraction.

II. RELATED WORK

Revisiting Iris Recognition with Color Cosmetic Contact Lenses

Above the years, iris recognition has full-fledged significance in the biometrics applications and is being used in numerous large measure nationally projects. Though iris patterns are inimitable, they may be exaggerated by exterior factors such as illumination, camera-eye direction, and sensor interoperability. The presence of contact lens, mainly color cosmetic lens, may also posture a experiment to iris biometrics as it complicates the iris patterns and deviate the inter and intraclass disseminations. This paper presents an in-depth study of the effect of contact lens on iris recognition concert. We also present the IIIT-D Contact Lens Iris database with over 6500 images relating to 101 subjects. For each subject, images are bagged without lens, translucent (prescription) lens, and color cosmetic lens (textured) using two different iris sensors. The results calculated using VeriEye recommend that color cosmetic lens expressively increases the false rejection at a fixed false reception rate. Also, the experiments on four prevailing lens detection algorithms suggest that incorporating lens detection helps in sustaining the iris recognition presentation. However, advance research is required to build sophisticated lens detection algorithm that can intensification iris recognition.

Unraveling the Effect of Textured Contact Lenses on Iris Recognition

The occurrence of a contact lens, predominantly a textured cosmetic lens, postures a task to iris recognition as it obscures the regular iris patterns. The main influence of this paper is to present an in-depth examination of the consequence of contact lenses on iris recognition. Two databases, specifically, the IIIT-D Iris Contact Lens database and the ND-Contact Lens database, are equipped to examine the differences triggered due to contact lenses. We also present a novel lens discovery algorithm that can be used to decrease the consequence of contact lenses. The future approach outdoes new lens detection algorithms on the two databases and displays enhanced iris recognition performance.

IV. PROPOSED WORK

Fourier transform to sense periodic phony iris patterns that were predominant in textured lenses mass-produced at that time. IRIS is one of the greatest auspicious biometric modalities, and is in consistent use in large-scale applications such as UAE port of entrance and India's UIDAI (Aadhar) schemes. Median filters, which effect the circulations of the bits to recognize the Hamming distance of phase. Tiring of contact lenses, both soft contacts and textured "cosmetic" soft contacts, damages the correctness of iris recognition. Our post-processing practices are Standardization, Segmentation using phase-based, texture analysis approaches.

Advantages:

By this practice, the consumer will have the sole identification for his private particulars. Two dissimilar contact lens iris image datasets have been together, self-reliantly in altered countries, using dissimilar iris sensors and sampling dissimilar brands of contact lenses. We can have extra effectiveness and safekeeping to the applications and also there will less memorial practice while storing in the data base. Instead of storage the iris image, the iris code is going to stock in the database.

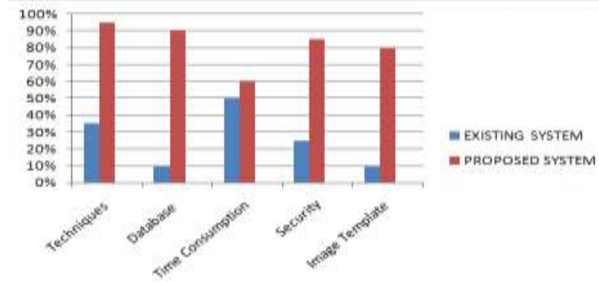

Fig.1.Comparision Of Existing System and Proposed System

Table 1: Comparision Of Existing System and Proposed System

	Techniques	Database	Time Consumption	Security	Image Template
EXISTING SYSTEM	35%	10%	50%	25%	10%
PROPOSED SYSTEM	95%	90%	60%	85%	80%

V. SYSTEM ANALYSIS

V.1 ALGORITHM SPECIFICATION:

V.1.1 GRAYSCALE:

Grayscale images are different from one-bit bi-tonal black-and-white images, which in the perspective of computer imaging are images by merely two colors, black, and white (also named bi-level or binary images). Grayscale images have numerous shades of gray in among.

Grayscale images are frequently the result of determining the intensity of light at every pixel in a distinct band of the electromagnetic spectrum, and in such cases they are monochromatic proper when only a given frequency is taken. But also they can be produced from a full color image; see the section about

renovating to grayscale.

```
for(int i = 0; i < img.getWidth(null); i++)
{
for(int j = 0; j < img.getHeight(null); j++)
{
clr_Tmp = new Color(int_Dar_AllPix[(j * img.getWidth(null)) + i]);
//Compute the grayscale of the color with the weights given (these are well documented)
int_Arr_Pixels[i][j] = (int)(0.3 * clr_Tmp.getRed() + 0.59 * clr_Tmp.getGreen() + 0.11 *
```

```
clr_Tmp.getBlue());
}
}
```

V.1.2 Median filter:

Usage of a median filter to develop an image harshly besmirched by faulty pixels.it is frequently necessary to be able to make some kind of noise lessening on an image or signal. The median filter is a nonlinear digital filtering technique, frequently used to eradicate noise. Such noise lessening is a typical pre-processing step to develop the results of future processing (for example, edge detection on an image). Median filtering is very extensively used in digital image processing because, under firm conditions, it conserves edges while eliminating noise.

```
allocate outputPixelValue[image width][image height]
allocate window[window width * window height]
edgex := (window width / 2) rounded down
edgey := (window height / 2) rounded down
for x from edgex to image width - edgex
for y from edgey to image height – edgey
i = 0
for fx from 0 to window width
for fy from 0 to window height
window[i] := inputPixelValue[x + fx - edgex][y + fy - edgey]
i := i + 1
sort entries in window[]
 outputPixelValue[x][y] := window[window width * window height / 2]
```

V.1.3 Edge Detection:

Edges portray confines and are therefore a problematic of essential reputation in image processing. Edges in images are areas with sturdy intensity differences – a jump in concentration from one pixel to the succeeding. Edge detecting an image knowingly diminishes the amount of data and sieves out unusable information, while conserving the significant structural possessions in an image. This was also stated in my Sobel and Laplace edge recognition class, but I just hunted reemphasize the point of why you would want to identify edges.

The Canny edge recognition algorithm is identified to numerous as the optimal edge detectors. Canny's intentions were to improve the many edge detectors previously out at the time he ongoing his effort. He was precise fruitful in attaining his goal and his ideas and approaches can be establish in his paper, "A Computational Method to Edge Detection". In his broadsheet, he shadowed a list of principles to develop existing approaches of edge detection. The first and best evident is low fault rate. It is significant that edges happening in images should not be unexploited and that there be NO retorts to non-edges. The second principle is that the edge points be well confined. In other disputes, the space between the edge pixels as originate by the detector and the definite edge is to be at a minutest. A third principle is to have only one reply to a single edge. This was executed because the first 2 were not considerable enough to entirely eradicate the probability of numerous responses to an edge.

Built on these standards, the canny edge detector first smooth’s the image to eradicate and noise. It then treasures the image ascent to highpoint areas with high spatial consequences. The algorithm then trails along these provinces and overwhelms any pixel that is not at the extreme (non-maximum suppression). The gradient array is currently further condensed by hysteresis. Hysteresis is used to trail laterally the enduring pixels that have not been suppressed. Hysteresis uses two thresholds and if the magnitude is beneath the first threshold, it is set to zero (made a non-edge). If the magnitude is overhead the high threshold, it is made an edge. Then if the magnitude is among the 2 thresholds, then it is fixed to zero except there is a trail from this pixel to a pixel with a gradient overhead T2.

Step 1

In order to proceed the canny edge detector algorithm, a sequence of steps must be trailed. The first step is to mesh out any noise in the unique image earlier trying to find and perceive any edges. And since the Gaussian filter can be figured using a modest mask, it is used entirely in the Canny algorithm. Once a appropriate mask has been intended, the Gaussian smoothing can be achieved using standard convolution techniques. A convolution mask is typically much smaller than the real image. As a result, the mask is glided over the image, deploying a square of pixels at a time. The greater the thickness of the Gaussian mask, the lesser is the detector's sensitivity to noise. The localization fault in the perceived edges also surges slightly as the Gaussian width is improved. The Gaussian mask used in my execution is shown below.

$\frac{1}{115}$ ×

2	4	5	4	2
4	9	12	9	4
5	12	15	12	5
4	9	12	9	4
2	4	5	4	2

Fig.2. Discrete approximation to Gaussian function with σ=1.4

Step 2

Later smoothing the image and eradicating the noise, the next step is to discovery the edge strength by enchanting the gradient of the image. The Sobel operator achieves a 2-D spatial gradient extent on an image. Then, the approximate absolute incline magnitude (edge strength) at each point can be established. The Sobel operator practices a pair of 3x3 complication masks, one approximating the gradient in the x-direction (columns) and the other estimating the gradient in the y-direction (rows). They are shown below:

Gx

-1	0	+1
-2	0	+2
-1	0	+1

Gy

+1	+2	+1
0	0	0
-1	-2	-1

Fig .3. (3x3) convolution masks

The magnitude, or EDGE STRENGTH, of the gradient is then approximated using the formula: |G| = |Gx| + |Gy|

Step 3

Finding the edge route is insignificant once the slope in the x and y directions are known. However, you will generate an mistake whenever sum X is identical to zero. So in the code there has to be a constraint set when this takes place. Whenever the slope in the x direction is identical to zero, the edge route has to be equivalent to 90 degrees or 0 degrees, liable on what the value of the slope in the y-direction is equivalent to. If GY has a value of zero, the edge route will be equivalent 0 degrees. Else the edge route will be equivalent to 90 degrees. The formula for verdict the edge direction is just: theta = invtan (Gy / Gx)

Step 4

When the edge route is known, the following step is to relate the edge route to a direction that can be outlined in an image. So if the pixels of a 5x5 image are ranged as follows:

x x x x x
x x x x x
x x a x x
x x x x x
x x x x x/

Then, it can be perceived by observing at pixel "a", there are only four likely directions when relating the surrounding pixels- 0 degrees (in the horizontal path), 45 degrees (lengthways the positive diagonal), 90 degrees (in the vertical path), or 135 degrees (lengthways the negative diagonal). So nowadays the edge alignment has to be determined into one of these four directions liable on which direction it is contiguous to (e.g. if the position angle is originate to be 3 degrees, make it nil degrees). Think of this as captivating a semicircle and dividing it into 5 sections. Thus, any edge track dwindling inside the yellow range (0 to 22.5 & 157.5 to 180 degrees) is set to 0 degrees. Some edge track dwindling in the green range (22.5 to 67.5 degrees) is fixed to 45 degrees. Any edge track dwindling in the blue range (67.5 to 112.5 degrees) is fixed to 90 degrees. And lastly, any edge track dwindling within the red range (112.5 to 157.5 degrees) is set to 135 degrees.

Step 5

Later the edge directions are known, Non-maximum suppression now has to be functional. Non-maximum suppression is used to trace lengthways the edge in the edge direction and suppress some pixel value (sets it equal to 0) that is not considered to be an edge. This will provide a thin line in the output image.

V.1.4 Pupil Detection:

Reference:https://opencv-code.com/tutorials/pupil-detection-from-an-eye-image/

```
for (int i = 0; i < contours.size(); i++)
{
double area = cv::contourArea(contours[i]);    // Blob area
cv::Rect rect = cv::boundingRect(contours[i]); // Bounding box
```

```
int radius = rect.width/2; // Approximate radius // Look for round shaped blob
if (area >= 30 &&
std::abs(1 - ((double)rect.width / (double)rect.height)) <= 0.2 &&
std::abs(1 - (area / (CV_PI * std::pow(radius, 2)))) <= 0.2)
{
cv::circle(src, cv::Point(rect.x + radius, rect.y + radius), radius, CV_RGB(255,0,0), 2);
}}
```

V.1.5 Hough transforms:

The Hough transform is a feature abstraction method used in image scrutiny, computer vision, and digital image processing. The determination of the method is to find defective cases of objects within a firm class of shapes by a voting technique. This voting technique is carried out in a parameter space, from which object candidates are gained as local maxima in a so-called accumulator space that is explicitly erected by the algorithm for computing the Hough transform.

V.1.6 Normalization:

Most normalization techniques are founded on transmuting iris into polar coordinates, recognized as unwrapping process. Pupil boundary and limbus boundary are usually two non-concentric outlines. The non-concentric state leads to different choices of reference points for altering an iris into polar coordinates. Proper choice of reference point is very significant where the radial and angular evidence would be distinct with esteem to this point.

V.1.7 Feature extraction:

When the input data to an algorithm is too huge to be administered and it is supposed to be disreputably redundant then the input data will be distorted into a reduced representation set of features. Converting the input data hooked on the set of features is called feature extraction. Uncertainty the features extracted are prudently chosen it is expected that the features set will extract the pertinent information from the input data in order to perform the anticipated task using this reduced depiction instead of the full extent input.

V.1.8 Hamming distance:

The Hamming distance among two strings of equivalent length is the number of positions at which the equivalent symbols are dissimilar. In another way, it measures the minimum number of replacements required to change one string into the other, or the minimum number of faults that could have distorted one string into the other.

V.1.9 MD5 Algorithm:

MD5 is an algorithm that is used to prove data integrity through the making of a 128-bit message digest from data input (which might be a message of some length) that is demanded to be as exclusive to that specific data as a fingerprint is to the specific distinct.

The main MD5 algorithm controls on a 128-bit state, divided hooked on four 32-bit words, meant A, B, C, and D. These are primed to certain static constants. The main algorithm then practices each 512-bit message block in turn to alter the state. The dispensation of a message block contains four alike stages, termed rounds; each round is composed of 16 alike operations based on a non-linear function F, modular addition, and left spin.

There are four likely functions F; a diverse one is used in every round:

MD5 Algorithm Description

We begin by assuming that we have a b-bit message as input, and that we demand to find its message digest. Here b is an random nonnegative integer; b may be zero, it need not be a several of eight, and it may be randomly large. We envisage the bits of the message written down as follows:

m_0 m_1 ... m_{b-1}

The following five steps are achieved to compute the message digest of the message.

Step1. Append Padding Bits

The message is "padded" (protracted) so that its size (in bits) is congruent to 448, modulo 512. That is, the message is prolonged so that it is just 64 bits shy of presence a numerous of 512 bits long. Padding is constantly achieved, even if the length of the message is previously matching to 448, modulo 512.

Padding is achieved as follows: a single "1" bit is added to the message, and then "0" bits are added so that the length in bits of the padded message changes congruent to 448, modulo 512. In all, at least one bit and at most 512 bits are added.

Step2. Append Length

A 64-bit depiction of b (the length of the message previously the padding bits were added) is added to the result of the preceding step. In the improbable event that b is superior than 2^64, then only the low-order 64 bits of b are used. (These bits are added as

two 32-bit words and added low-order word first in agreement with the
Preceding pacts.)
At this point the resultant message (after stuffing with bits and with b) has a length that is an exact multiple of 512 bits. Consistently, this message has a length that is an meticulous multiple of 16 (32-bit) words. Let M[0 ... N-1] denote the words of the resultant message, everyplace N is a multiple of 16.

Step3. Initialize MD Buffer

A four-word buffer (A, B, C, and D) is used to calculate the message digest. Here every of A, B, C, D is a 32-bit register. These registers are prepared to the following values in hexadecimal, low-order bytes first):

Letter A: 01 23 45 67
Letter B: 89 ab cd ef
Letter C: fe dc ba 98
Letter D: 76 54 32 10

Step4. Process Message in 16-Word Block

We first describe four supplementary functions that all take as input three 32-bit words and create as output one 32-bit word.

F1 (X,Y,Z) = XY v not(X) Z
G1 (X,Y,Z) = XZ v Y not(Z)
H1 (X,Y,Z) = X xor Y xor Z
I1(X,Y,Z) = Y xor (X v not(Z))

In every bit position F1 turns as a conditional: if X then Y else Z. The function F might have been different using + in its place of v since XY and not(X)Z will never ensure 1's in the same bit point.) It is stimulating to note that if the bits of X, Y, and Z are independent and impartial, the each bit of F1(X,Y,Z) will be independent and impartial.

The functions G1, H1, and I1 are alike to the function F1, they act as "bitwise parallel" to create their output from the bits of X, Y, and Z, in such a way that if the consistent bits of X, Y, and Z are independent and impartial, then each bit of G1(X,Y,Z), H1(X,Y,Z), and I1(X,Y,Z) will be independent and impartial. Note that the function H1 is the bit-wise "xor" or "equality" function of its inputs.

This stage uses a 64-element table T[1 ... 64] built from the sine function. Let T[i] denote the i-th element of the table, which is equivalent to the integer part of 4294967296 times abs(sin(i)), wherever i is in radians.

Step5. Output

The message digest formed as output is A, B, C, D. That is, we initiate with the low-order byte of A, and finish with the high-order byte of D.
This concludes the explanation of MD5.

MD5 Hash Properties

The MD5 hash contains of a small amount of binary data, typically no more than 128 bits. All hash values share the following properties:

A. Hash length

The length of the hash value is unwavering by the type of the used algorithm, and its length does not hinge on the size of the file. The utmost common hash value lengths are one or the other 128 or 160 bits.

B. Non-discoverability

All pair of non-identical files will interpret into a completely dissimilar hash value, even if the two files vary only by a single bit. Using current technology, it is not likely to discover a pair of files that interpret to the similar hash value.

C .Repeatability

All time a specific file is hashed using the similar algorithm , the exact similar hash value will be formed.

D. Irreversibility

Wholly hashing algorithms are one-way. Assumed a checksum value, it is infeasible to determine the password. In fact, not a bit of the properties of the unique message can be resolute given the checksum value alone.

V.2 MODULE DESCRIPTION

V.2.1 Image Conversion

Grayscale images are different from one-bit black-and-white images, which in the perspective of computer imaging are images with merely the two colors, black, and white (also called binary images or bi-level image). Grayscale images have several shades of gray in amongst. Grayscale images are also termed monochromatic, signifying the nonappearance of any chromatic variation.
Grayscale images are frequently the result of computing the intensity of light at all pixel in a

solitary band of the electromagnetic spectrum (e.g. ultraviolet, infrared, visible light, etc.), and in such cases they are monochromatic proper when individually a given frequency is taken. But likewise they can be created from a full color image; see the section about changing to grayscale.

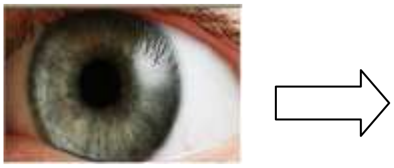

Fig. 4.1 Original Image

Fig. 4.2 Grayscale Image

V.2.2 Edge Detection

Edge detection is a necessary tool in computer vision and image processing, mainly in the areas of feature recognition and feature abstraction, which aim at recognizing points in a digital image at which the image brightness variations properly or, more sharply, has discontinuities. The edges detached from a two-dimensional image of a three-dimensional scene can be categorized as

[i] Viewpoint independent.

[ii] Viewpoint dependent.

A viewpoint independent edge characteristically reflects inherent assets of the three-dimensional objects, such as surface designs and surface shape.

A viewpoint dependent edge may vary as the viewpoint variations occurs, and characteristically reflects the geometry of the scene, such as objects obstructing one another

V.2.3 Pupil Detection

The attained iris image has to be preprocessed to spot the iris, which is an annular portion amongst the sclera (outer boundary) and the pupil (inner boundary).The first step in iris localization is to spot pupil which is the black circular part bounded by iris tissues. The center of pupil can be used to identify the outer radius of iris patterns. The vital steps involved are:

[i]Pupil detection (Inner Circle)

[ii]Outer iris localization

Circular Hough Transformation for pupil recognition can be used. The modest idea of this method is to determine curves that can be parameterized like, polynomials, circles, straight lines etc., in a suitable parameter space.

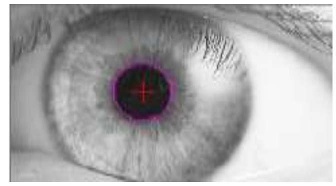

Fig.5. Detection of inner pupil boundary

External noise is detached by blurring the intensity image. But abundant blurring may dilate the borders of the edge or may make it tough to detect the outer iris boundary, separating the eyeball and sclera. Thus a distinctive smoothing filter such as the median filter is used on the actual intensity image. This kind of filtering removes sparse noise while preserving image borders. After filtering, the contrast of image is enhanced to have severe variation at image borders using histogram equalization.

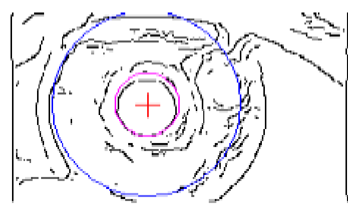

Fig.6. Detection of inner pupil boundary

V.2.4 Normalization

We must eliminate blurred images before feature extraction. Localizing iris from an image delineates the annular portion from the rest of the image. The idea of rubber sheet model recommended by Daugman takes into consideration the probability of pupil dilation and appearing of dissimilar size in different images. For this purpose, the coordinate system is transformed by un-wrapping the iris and mapping all the points inside the border of the iris into their polar alike. The mapped image has 80 × 360 pixels. The aforementioned means that the step size is alike at all angles. This normalization somewhat diminishes the elastic distortions of the iris.

V.2.5 Feature extraction

Corners in the normalized iris image can be reprocessed to extract features for characteristic two iris images. The phases involved in corner detection algorithm are as follows

S1: The normalized iris image is used to spot corners using covariance matrix

S2: The sensed corners between the database and query image are used to discover cross correlation coefficient

S3: If the amount of correlation coefficients amongst the sensed corners of the two pictures is larger than a threshold value then the candidate is recognized by the system.

V.2.6 Matching

Two irises are determined to be of the similar class by a association of the feature vectors, using a Daugman like X-OR operation. Lastly matching would be done of the iris. The similarity would be done with the trained pictures. If the images are coordinated and existing in our database it shows the particulars of that person. If he is not coordinated with the database, then his details will be collected for advance investigation, if it is required.

VI. FUTURE WORKS and CONCLUSION

There are numerous reasons to be certain of that biometrics will transform the life of people in near future commonly because its use will be much more suitable than other techniques in use today for individual uniqueness authentication. This is already obvious today, especially in connection with applications such as physical and reasonable access control, transportation, and also in the financial productiveness.

In this effort, we have discovered a method of creating iris textures for a given person embedded in their regular iris texture (or somebody else's if anticipated) using just the iris code of the person. If these textures are reprocessed in an iris recognition system, they will give a answer alike to the unique iris texture. There are some papers that deliberate the formation of artificial iris textures using cues from anatomy, or by demonstrating iris textures using various mathematical models from a unadulterated synthesis point of view. To the finest of our knowledge, no work currently exist that starts demonstrating the iris from the iris code which is usually considered to be unidentifiable data. In our work, we generate the iris texture begin from just the iris bit code of the individual and we embed the required texture to generate a iris code. Our outcomes show ordinary looking iris pictures that give a related recognition (verification) performance as a genuine iris of the same person. As stated in the offset of this section, the benefit of this is that we can now generate alternative iris textures that will give a very similar iris code when associated to the original iris. As upcoming work, we will discover countermeasures for spotting such hard work.

REFERENCES

[1] P .Vijayaraagavan, R.C Narayanan "Web Image Re-Ranking Using Hash Based Signature"SSRG-IJCSE-V213P115.

[2] P.Kalavathi,"A Thresholding Method for Color Image Binarization" SSRG-IJCSE-V117P107

[3] R. Connaughton, A. Sgroi, K. Bowyer, and P. Flynn, "A multi algorithm analysis of three Iris biometric sensors," IEEE Trans. Inf. Forensics Security, vol. 7, no. 3, pp. 919–931, Jan. 2012.

[4] J. J. Nichols. (2012). Annual Report: Contact Lenses [Online]. Available: http://www.clspectrum.com/articleviewer.aspx?articleID=107853

[5] N. Kohli, D. Yadav, M. Vatsa, and R. Singh, "Revisiting Iris recognition with color cosmetic contact lenses," in Proc. 6th IAPR, 2013, pp. 1–5.

[6] M. Negin, T. Chmielewski, M. Salganicoff, U. von Seelen, P. Venetainer, and G. Zhang, "An Iris biometric system for public and personal use," IEEE Compute., vol. 33, no. 2, pp. 70–75, Feb. 2000.

[7] J. W. Thompson, H. Santos-Villalobos, P. J. Flynn, and K. W. Bowyer, "Effects of Iris surface curvature on Iris recognition," in Proc. 6th IEEE Int. Conf. Biometrics, Technol., Appl., Syst., 2013, pp. 1–3.

[8] S. Baker, A. Hentz, K. Bowyer, and P. Flynn, "Degradation of Iris recognition performance due to non-cosmetic prescription contact lenses," Compute. Vis. Image Understand., vol. 114, no. 9, pp. 1030–1044, 2010.

[9] J. Daugman, "Demodulation by complex-valued wavelets for stochastic pattern recognition," Int. J. Wavelets, Multiresolution Inf. Process., vol. 1, no. 1, pp. 1–17, 2003.

[10] E. C. Lee, K. R. Park, and J. Kim, "Fake Iris detection by using purkinje image," in Proc. IAPR Int. Conf. Biometrics, 2006, pp. 397–403.

[11] X. He, S. An, and P. Shi, "Statistical texture analysis-based approach for fake Iris detection using support vector machines," in Proc. IAPR Int. Conf. Biometrics, 2007, pp. 540–546.

[12] Z. Wei, X. Qiu, Z. Sun, and T. Tan, "Counterfeit Iris detection based on texture analysis," in Proc. 18th Int. Conf. Pattern Recognit., 2008, pp. 1–4.

[13] Z. He, Z. Sun, T. Tan, and Z. Wei, "Efficient Iris spoof detection via boosted local binary patterns," in Advances in Biometrics. New York, NY, USA: Springer-Verlag, 2009, pp. 1080–1090.

AUTHORS

First Author: Mrs. GATHEEJATHUL KUBRA .J, PG Student, Mohamed Sathak A.J College Of Engineering, Siruseri, Chennai, Tamil Nadu, India, Second Author: Mr. RAJESH P, HOD, Computer science Department, Mohamed Sathak A.J College of Engineering, Siruseri, Chennai, Tamil Nadu, India,