Robust Variable Embedding Retinal Biometric Steganography and Authentication Checking

Pabak Indu¹, Souvik Bhattachryya²

¹ School of Computers, Inspiria Knowledge Campus, Siliguri, West Bengal, India
² Department of Computer Science and Engineering, University Institute of Technology, The University of Burdwan, Burdwan, India

Abstract: Attacks, misuse or unauthorized access of information is of great concern today, which makes the protection of documents through digital media a priority problem. The Biometric Steganography plays a vital role in maintaining the secrecy of the secret information in today’s world of communication. Although the biometric system along is always at the risk of suffering the possibility of different types of attacks. The super imposition of biometric system in steganography principle has drastically cuts down those potential risks. This paper has proposed a new approach of information hiding methodology in the area of retinal biometric steganography in combination with biometric authentication principle. Experimental results demonstrate the effectiveness and accuracy of the proposed methodology in terms of security of hidden data with maintaining the integrity and authenticity of it.

Keywords: Authentication checking, Retinal Biometric Steganography, Pixel Selection, Feature Extraction, Polynomial value digitation.

I. INTRODUCTION

In current digitized world humane race is facing a great revolution over Internet Technologies where every single information needs to be transmitted securely through some communication media.

This results the birth of different Information Hiding methodologies like, Steganography, Cryptography, Watermarking etc. [1]. The first two techniques provide protection on data whereas the third one gives the authentication of data. Steganography is an area of information hiding which means "secret or covered writing". Almost all digital file formats can be used for steganography, but the image and audio files are more suitable because of their high degree of redundancy [2].

Biometric security system is an automatic recognition system of an individual through the aid of the physiological and behavioural [3]–[7] characteristic. The term "biometrics" is a Greek word where the word “bio” means “life” and “metric” means “to measure”. This system finds out the person's uniqueness based on pattern analyses, carried out on unambiguous human behaviour [4], [5]. Physiological biometric systems consist of fingerprints, retina, iris, hand geometry, hand vein, ear shape and facial recognition systems [6] where the features are usually unalterably processed by human being.

In the contrary, the behavioural biometric characteristics are valid over a short span of time. Examples of behavioral biometric systems are voice recognition, keystroke dynamics, signature verification and gait analysis [7]. Biometrics refers to the automatic identification of a person based on his or her physiological or behavioural characteristics.

This identification method is preferred over traditional methods involving passwords and personal identification numbers (PINs). Uses of biometric techniques has been gaining popularity day-by-day due its features like Uniqueness, Universality, Performance, Measurability and User friendliness.

In this work a specific image based biometric steganography method has been proposed which may be considered as the improved version of the author’s previous work [8] with an additional approach of authentication of the secret message, key or password embeds into transform domain portions of cover image with the help of the biometric features. This developed contribution is a novel biometric steganography technique, where the secret message hiding is done in the Transform domain with variable length embedding technique as a bit stream of two to four bits combinations and authentication checking is done in the spatial domain of the same image with the aid of some biometric retinal features.

Rest of the paper has been organized as following sections: Section II describes some associated works on image steganography. Section III deals with proposed method. Section IV and V has been used for describing the Algorithms and Mathematical Analysis of the proposed system respectively. In section VI experimental results has been discussed and finally section VII draws the conclusion.
II. RELATED WORK

In this section some image based steganography data hiding methods in both spatial domain and transform domain has been discussed. This section also discusses some existing Biometric Security Techniques.

A. Image Steganography Techniques:

The image steganography can be designed and implemented in two domains, one is Spatial and another one is in transform domain. There are numerous approaches exists in both domains. Some of them are discussed below:

1) Spatial Domain Technique: The most common technique in this domain is data hiding by least-significant-bit (LSB) [9] which works based on manipulating the least-significant bit (LSB) planes by directly replacing the LSBs of the cover-image with the message bits. The pixel-value differencing (PVD) method proposed by Wu and Tsai [10] can successfully provide both high embedding capacity and outstanding imperceptibility for the stego-image. In 2004, Potdar et al. [11] proposes GLM (Gray level modification) technique which is used to map data by modifying the gray level of the image pixels. Hong and Chen [12] introduced a new method based on pixel pair matching (PPM). Bhattacharyya and Sanyal proposed a new image transformation technique known as Pixel Mapping Method (PMM) [13], [14] a method for information hiding within the spatial domain of any gray scale image. Data embedding are done by mapping each two or four bits of the secret message in each of the neighbor pixels based on some features of that pixel. Banerjee et al. [15] proposes Pixel Factor Mapping (PFM) technique which embeds the four bits of secret message in a single pixel intensity based on the maximum prime factor value of pixel intensity.

2) Transform Domain Technique: Transform domain steganography method hides messages in significant areas of cover image which makes them robust against various image processing operations like compression, enhancement etc. The widely used transformation functions include Discrete Cosine Transformation (DCT), Discrete Fourier Transform (DFT), and Wavelet Transformation.

3) DCT based Data Hiding: J-Steg [16] and JP Hide [17] are the two classical JPEG steganography tools developed based on LSB embedding technique. F5 steganography algorithm was introduced by Westfeld [18] where instead of replacing the LSBs of quantized DCT coefficients with the message bits, it modifies the randomly-chosen coefficient by decreasing the absolute value of the coefficient by one. Out Guess [19] has been developed through UNIX. Yet Another Steganography Scheme (YASS) [20] works based on the principle of JPEG steganography but does not directly embed data in JPEG DCT coefficients. Instead an input image in spatial domain is divided into blocks with a fixed large size known as the big blocks (or B-blocks). MB steganography methods has been proposed for JPEG images, achieves a high embedding efficiency and message capacity than the previous methods also remains secure against first order statistical attacks. BCH and BCHopt [21] are side-informed algorithms that employ BCH codes to minimize the embedding distortion in the DCT domain defined using the knowledge of non-rounded DCT coefficients. Wang et al. [22] presents an efficient JPEG steganography scheme based on the block entropy of DCT coefficients and syndrome trellis coding (STC). Bhattacharyya et al. [23] introduced DCTDM, which can embed via modulating adjacent DCT coefficient differences.

2) DWT based Data Hiding: Wavelet-based steganography [24] and [25] is a new idea in the application of wavelets. However, the standard technique of storing in the least significant bits (LSB) of a pixel still applies. The only difference is that the information is stored in the wavelet coefficients of an image, instead of changing bits of the actual pixels.

B. Biometric Information Security

Biometrics system intends to recognize an individual through physiological or behavioral attributes, for instance face, fingerprint, iris, retina and DNA also [26]. In biometric technique there are various ways and all biometric techniques differ according to security level, user acceptance, cost and performance. Fig. 1 describes the classification of biometric techniques.

222Fingerprint: Fingerprint [27] is one of the techniques which provide biometric securities, based on fingerprint pattern recognition. There are three basic patterns of fingerprint ridges: i) Arch: Ridges enter from one side of the finger, forming in the center and exit the other side of the finger. ii) Loop: Ridges enter from one side of a finger then form a curve and then exit on that same side. iii) Whorl: The ridges form circularly around a central point on the finger.

1) Retina: Analyzing the complex structure of the capillaries that is the layer of blood vessels at retina which is not entirely genetically determined i.e. back of eye is involved in this procedure [28].

2) Face: Face biometry [29] depends on analyzing facial characteristics. It is automatically identifying or verifying a person from an image.
3) **Hand Geometry**: In this mechanism [30] the shape of the human hand is computed and analyzed is based on the palm and fingers structure, width of the fingers in different places, length of the fingers, thickness of the palm area, etc.

4) **Nose**: The nose biometric technique [31] works through features extracting from a nose and by the help of various classification techniques.

5) **Ear**: One of interesting authentication technique is ear biometric security. Analyzing ear shape and area measurement of a human can be identified easily [32]

6) **Signature**: Signature signing features like writing speed, velocity and pressure are used for identifications. Signature verification devices are logically accurate in operation and lend themselves to applications where a signature is an accepted identifier [33]. It can be operated in two different ways like Static and Dynamic

7) **Iris**: In this iris-based biometric system [34] features are analyzed using mathematical pattern-recognition techniques. It stores the measurement of the colored ring of tissue surrounds the pupil of eye.

8) **Voice**: Voice biometrics [35] has the most probable for enlargement, because it requires no new hardware, most PCs have already contained a microphone.

9) **Vein geometry**: In this technique the vein of hand, vein of finger, vein of palm etc. are used for authentication purpose. L. Wang et al. [36] proposed a verification system of human beings using the thermal-imaged vein pattern in the back of hand. A. Kumar et al. [37] presents a technique which can authenticate a person based on minutiae matching of vein junction points.

10) **Face geometry**: One of the physiological characteristics is Face geometry [38] for recognition of a human. The relative location of human face objects like mouth, eye, nose etc in the face is unique for each human being. Face length, height, width, curvatures, relative location like distance and angles of facial objects has been observed in the face geometry of a human face.

### III. PROPOSED METHOD

In this work a new approach of biometric steganography technique has been proposed which works in a combined approach like transform and spatial domain in a same image. This biometric steganography technique has been designed for embedding secret message in the skin tone portion of the human face using variable length message embedding algorithm in transform domain with the help of discrete cosine transform and biometric features of retinal images like Rod and Cone count for authenticity checking, in spatial domain. The block diagram of the proposed system has been shown in Fig. 2. The system first performs biometric features embedding in spatial domain then performs embedding in transform domain which divides the system architecture in two parts

#### A. Biometric Features Embedding in Spatial Domain

This architecture consists of the steps as depicted in fig. 2 and block diagram of each steps are given below.

1) **Step 1: Load Cover Image** – In this step the cover image is loaded into the system and border cells are cropped.
2) **Step 2: Skin Tone Detection** – In this step the skin tone area detection in the input cover image is performed by setting some threshold value on hue plane and saturation plane of HSV color space.

![Block Diagram of Proposed System](image1)

3) **Step 3: Pixel Selection for Biometric feature embedding in Spatial Domain** – In this step the system select those pixels for spatial domain embedding out of pixels selected through skin tone detection by dividing the whole image in 5 X 5 blocks and selecting pixels on every first row and first column of each block only on skin tone detected area.

In this work the skin region of the image has been used for embedding of secret data as well as biometric features. The system detects the skin color with the help of skin detector and skin classifier. Skin detector has been used to convert the RGB color space into appropriate color space HSV, as because it is more appropriate for human colour perception. For skin detection threshold value has been chosen for Hue as well as Saturation range. Next skin classifier classifies the pixels of the cover image to skin and non-skin pixels by defining a boundary. The skin detection algorithm produces a mask which is simply a black and white pixel with the help of threshold, which has a predefined range associated with the target skin pixel values.

![Block Diagram of Skin Tone Detection](image2)
4) **Step4: Embedding Pixel Selection** – Pixel for embedding space has been selected with the help of two square series having above next and below next value of the concern pixel value. In the next step it calculates the mid value of those two square series. If the resulting mid value is greater than the pixel value then embedding pixel value is calculated by subtracting the concern pixel value from the below next square series value of the pixel. Else if the resulting mid value is lower than the concern pixel value then embedding pixel value is calculated by subtracting the concern pixel value from the above next square series value of the pixel.

5) **Step5: Embedding through variable bit length** – Based upon the availability of embedding space the message bit stream is embedded in the resulting polynomial value format.

The embedding polynomial used here is as follows:

\[ \sum_{n=1}^{N} n \cdot (k + \text{bit}_n) + \lambda n \cdot (k + \text{bit}_{n-1}) + (\text{x} \cdot n \cdot (k + \text{bit}_{n-2}) + \ldots + (\text{x} \cdot (k + \text{bit}_1)) \],

where, \( \text{bit}_n \) is the \( n \text{th} \) bit of remainder bit stream and \( k \) be a value which classify the polynomial value for different length bit stream. Thus for different \( k \) values the polynomial generates different class values for different length bit stream in the following manner.
Thus the system embed the bit stream according to every pixels embedding space $\delta_0$ and if there is not enough embedding space $\delta_j$ then the system go for next pixel of the plane and if data is fully embedded in that particular plane then the system perform embedding in next plane. Table I below explain the embedding principle for $k=1$.

**Table I**

<table>
<thead>
<tr>
<th>Bit Stream</th>
<th>Polynomial</th>
<th>Polynomial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2(1+0)</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>2(1+1)</td>
<td>4</td>
</tr>
<tr>
<td>00</td>
<td>$2^2(1+0)+2(1+0)$</td>
<td>6</td>
</tr>
<tr>
<td>01</td>
<td>$2^2(1+0)+2(1+1)$</td>
<td>8</td>
</tr>
<tr>
<td>10</td>
<td>$2^2(1+1)+2(1+0)$</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>$3^2(1+1)+2(1+1)$</td>
<td>12</td>
</tr>
<tr>
<td>000</td>
<td>$2^3(1+0)+2^2(1+0)+2(1+0)$</td>
<td>14</td>
</tr>
<tr>
<td>001</td>
<td>$2^3(1+0)+2^2(1+0)+2(1+1)$</td>
<td>16</td>
</tr>
<tr>
<td>010</td>
<td>$2^3(1+0)+2^2(1+1)+2(1+0)$</td>
<td>18</td>
</tr>
<tr>
<td>011</td>
<td>$2^3(1+0)+2^2(1+1)+2(1+1)$</td>
<td>20</td>
</tr>
<tr>
<td>100</td>
<td>$2^3(1+1)+2^2(1+0)+2(1+0)$</td>
<td>22</td>
</tr>
<tr>
<td>101</td>
<td>$2^3(1+1)+2^2(1+0)+2(1+1)$</td>
<td>24</td>
</tr>
<tr>
<td>110</td>
<td>$2^3(1+1)+2^2(1+1)+2(1+0)$</td>
<td>26</td>
</tr>
<tr>
<td>111</td>
<td>$2^3(1+1)+2^2(1+1)+2(1+1)$</td>
<td>28</td>
</tr>
<tr>
<td>0000</td>
<td>$2^4(1+0)+2^3(1+0)+2^2(1+0)+2(1+1)$</td>
<td>30</td>
</tr>
<tr>
<td>0001</td>
<td>$2^4(1+0)+2^3(1+0)+2^2(1+0)+2(1+1)$</td>
<td>32</td>
</tr>
<tr>
<td>0010</td>
<td>$2^4(1+0)+2^3(1+0)+2^2(1+1)+2(1+0)$</td>
<td>34</td>
</tr>
<tr>
<td>0011</td>
<td>$2^4(1+0)+2^3(1+0)+2^2(1+1)+2(1+1)$</td>
<td>36</td>
</tr>
<tr>
<td>0100</td>
<td>$2^4(1+0)+2^3(1+1)+2^2(1+0)+2(1+0)$</td>
<td>38</td>
</tr>
<tr>
<td>0101</td>
<td>$2^4(1+0)+2^3(1+1)+2^2(1+0)+2(1+1)$</td>
<td>40</td>
</tr>
<tr>
<td>0110</td>
<td>$2^4(1+0)+2^3(1+1)+2^2(1+1)+2(1+0)$</td>
<td>42</td>
</tr>
<tr>
<td>0111</td>
<td>$2^4(1+0)+2^3(1+1)+2^2(1+1)+2(1+1)$</td>
<td>44</td>
</tr>
<tr>
<td>1000</td>
<td>$2^4(1+1)+2^3(1+0)+2^2(1+0)+2(1+0)$</td>
<td>46</td>
</tr>
<tr>
<td>1001</td>
<td>$2^4(1+1)+2^3(1+0)+2^2(1+0)+2(1+1)$</td>
<td>48</td>
</tr>
<tr>
<td>1010</td>
<td>$2^4(1+1)+2^3(1+0)+2^2(1+1)+2(1+0)$</td>
<td>50</td>
</tr>
<tr>
<td>1011</td>
<td>$2^4(1+1)+2^3(1+0)+2^2(1+1)+2(1+1)$</td>
<td>52</td>
</tr>
<tr>
<td>1100</td>
<td>$2^4(1+1)+2^3(1+1)+2^2(1+0)+2(1+0)$</td>
<td>54</td>
</tr>
<tr>
<td>1101</td>
<td>$2^4(1+1)+2^3(1+1)+2^2(1+0)+2(1+1)$</td>
<td>56</td>
</tr>
<tr>
<td>1110</td>
<td>$2^4(1+1)+2^3(1+1)+2^2(1+1)+2(1+0)$</td>
<td>58</td>
</tr>
<tr>
<td>1111</td>
<td>$2^4(1+1)+2^3(1+1)+2^2(1+1)+2(1+1)$</td>
<td>60</td>
</tr>
</tbody>
</table>
6) **Step 6: Filling up Non Embedding Portions:** Remaining non embedding portions is filled by the system with respect to the selected square series value of that pixel. Since the embedding polynomial value is always even number for spatial domain thus if the selected square series value is odd number then the pixel value is made to an even number for denoting that no data has been embedded there.

7) **Step 7: Stego Image Creation** – Resulting Stego image for Spatial Domain is created.

**B. Secret Message Hiding in Transform Domain:**

The architecture consists of the steps which is depicted in below figures.

1) **Step 1: Load Cover Image** – In this step the cover image is loaded into the system and border cells are cropped.

2) **Step 2: Skin Tone Detection** – Already explained earlier.

3) **Step 3: Pixel Selection for Secret Message Hiding in Transform Domain** – For hiding secret message in Transform domain, the procedure made a block processing on the resulting cropped image which is cropped into 5 X 5 blocks and take the 4 X 4 block of each 5 X 5 blocks by ignoring first row and first column, since spatial domain embedding operation is performed at those cells contained in first row and first column.

![Fig 7: Pixel Selection on each block](image)

4) **Step 4: Preprocessing for Embedding** – This steps performs two dimensional Discrete Cosine Transformation (DCT2) on 4 X 4 block of each 5 X 5 block by ignoring 1st row and 1st column. Then the system ignore topmost left corner cell value and extract the integer part of other cell values of the 4 X 4 block.

![Fig 8: Block Diagram of Preprocessing](image)

For hiding secret message in Transform domain it made a block processing on the resulting cropped image which divide the cropped image into 5 X 5 blocks and take the 4 X 4 block of each 5 X 5 blocks by ignoring first row and first column, since spatial domain embedding operation is performed at those cells contained in first row and first column.

| 998.2345 | 7.376 | 20.98 | 100.287 |
| 34.67 | -27.834 | 89.342 | 78.534 |
| 55.783 | 46.983 | -29.345 | 55.439 |
| 12.678 | 0.534 | 29.356 | 78.243 |

![Fig 9: Cell value extraction](image)
5) **Step5: Embedding Space Selection** – As the technique the system follows embedding space varies from pixel to pixel. To get the embedding space of a cell of each block the system uses a series of integers and thus the system adds the squared value of integers up to the below next value $\alpha_{ij}$ of the cell value $Cell_{ij}$ and also do the same up to the next above value $\beta_{ij}$ of the cell value $Cell_{ij}$. As for example, $Cell_{ij}$ = 127.

Then, $\alpha_{ij}=1^2+2^2+3^2+4^2+5^2+6^2=91 \,<\, 127$ and $\beta_{ij}=1^2+2^2+3^2+4^2+5^2+6^2+7^2=140 \,>\, 127$.

Then the system calculate the mid value by $\gamma_{ij}=\frac{(\alpha_{ij}+\beta_{ij})}{2}$ and if $Cell_{ij}$ is less than or equal to the mid value $\gamma_{ij}$ then it get the subtracted value $\delta_{ij}=Cell_{ij}-\alpha_{ij}$ else $\delta_{ij}=\beta_{ij}-Cell_{ij}$ as embedding space. Thus after getting the embedding space for each cell $Cell_{ij}$ the system will further approach for embedding.

6) **Step5: Embedding** – With respect to the availability of embedding space the message bit stream is embedded in the resulting polynomial value format.

The embedding polynomial used here is as follows:

$$\sum_{n=x}^{y}(k+bit_1)+x^{n-1}.(k+bit_2)+x^{n-2}.(k+bit_3)+............+x.(k+bit_n)$$

where, $bit_n=n^{th}$ bit of remainder bit stream and $k$ be a value which classify the polynomial value for different length bit stream. Thus for different $k$ values the polynomial generates different class values for different length bit stream.

a) **Step 5.1: Choosing value of the parameters of the polynomial with respect to embedding in Transform domain** – In Transform domain the embedded message is changed to the interval $[+2\,-\,2]$ at the extraction time, so at extraction time it is difficult extract the original message and this problem is overcome here by choosing the parameter values of the polynomial in such a manner so that it produces values with difference of 5 between each consecutive values. So that at extraction time the system will fetch the nearest value of the polynomial. Thus the system makes two halves of each polynomial value up to the range $+2$ and $-2$.

As for example, if the polynomial value 10 is change to $10-2=8$, then the system will automatically choose the nearest value 10 as it knows that the highest change can occur in the interval $[+2\,-\,2]$.

Similarly, if the polynomial value 10 is change to $10+1=11$ as the pointer point to in fig. 10 then the system will automatically choose the nearest value 10 as it knows that the highest change can occur in the interval $[+2\,-\,2]$. 

Fig 10: Value Fetching (Lower)

![Fig 10: Value Fetching (Lower)](image)

Fig 11: Value Fetching (Higher)

![Fig 11: Value Fetching (Higher)](image)
Table II below explain the embedding principle for x=5 and k=1.

**TABLE II**

<table>
<thead>
<tr>
<th>Bit Stream</th>
<th>Polynomial</th>
<th>Polynomial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5(1+0)</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>5(1+1)</td>
<td>10</td>
</tr>
<tr>
<td>00</td>
<td>5^2(1+0)+5(1+0)</td>
<td>30</td>
</tr>
<tr>
<td>01</td>
<td>5^2(1+0)+5(1+1)</td>
<td>35</td>
</tr>
<tr>
<td>10</td>
<td>5^2(1+1)+5(1+0)</td>
<td>55</td>
</tr>
<tr>
<td>11</td>
<td>5^2(1+1)+5(1+1)</td>
<td>60</td>
</tr>
<tr>
<td>000</td>
<td>5^3(1+0)+5^2(1+0)+5(1+0)</td>
<td>155</td>
</tr>
<tr>
<td>001</td>
<td>5^3(1+0)+5^2(1+0)+5(1+1)</td>
<td>160</td>
</tr>
<tr>
<td>010</td>
<td>5^3(1+0)+5^2(1+1)+5(1+0)</td>
<td>180</td>
</tr>
<tr>
<td>011</td>
<td>5^3(1+0)+5^2(1+1)+5(1+1)</td>
<td>185</td>
</tr>
<tr>
<td>100</td>
<td>5^3(1+1)+5^2(1+0)+5(1+0)</td>
<td>280</td>
</tr>
<tr>
<td>101</td>
<td>5^3(1+1)+5^2(1+0)+5(1+1)</td>
<td>285</td>
</tr>
<tr>
<td>110</td>
<td>5^3(1+1)+5^2(1+1)+5(1+0)</td>
<td>305</td>
</tr>
<tr>
<td>111</td>
<td>5^3(1+1)+5^2(1+1)+5(1+1)</td>
<td>310</td>
</tr>
<tr>
<td>0000</td>
<td>5^4(1+0)+5^3(1+0)+5^2(1+0)+5(1+0)</td>
<td>780</td>
</tr>
<tr>
<td>0001</td>
<td>5^4(1+0)+5^3(1+0)+5^2(1+0)+5(1+1)</td>
<td>785</td>
</tr>
<tr>
<td>0010</td>
<td>5^4(1+0)+5^3(1+0)+5^2(1+1)+5(1+0)</td>
<td>805</td>
</tr>
<tr>
<td>0011</td>
<td>5^4(1+0)+5^3(1+0)+5^2(1+1)+5(1+1)</td>
<td>810</td>
</tr>
<tr>
<td>0100</td>
<td>5^4(1+0)+5^3(1+1)+5^2(1+0)+5(1+0)</td>
<td>905</td>
</tr>
<tr>
<td>0101</td>
<td>5^4(1+0)+5^3(1+1)+5^2(1+0)+5(1+1)</td>
<td>910</td>
</tr>
<tr>
<td>0110</td>
<td>5^4(1+0)+5^3(1+1)+5^2(1+1)+5(1+0)</td>
<td>930</td>
</tr>
<tr>
<td>0111</td>
<td>5^4(1+0)+5^3(1+1)+5^2(1+1)+5(1+1)</td>
<td>935</td>
</tr>
<tr>
<td>1000</td>
<td>5^4(1+1)+5^3(1+0)+5^2(1+0)+5(1+0)</td>
<td>1405</td>
</tr>
<tr>
<td>1001</td>
<td>5^4(1+1)+5^3(1+0)+5^2(1+0)+5(1+1)</td>
<td>1410</td>
</tr>
<tr>
<td>1010</td>
<td>5^4(1+1)+5^3(1+0)+5^2(1+1)+5(1+0)</td>
<td>1430</td>
</tr>
<tr>
<td>1011</td>
<td>5^4(1+1)+5^3(1+0)+5^2(1+1)+5(1+1)</td>
<td>1435</td>
</tr>
<tr>
<td>1100</td>
<td>5^4(1+1)+5^3(1+1)+5^2(1+0)+5(1+0)</td>
<td>1530</td>
</tr>
<tr>
<td>1101</td>
<td>5^4(1+1)+5^3(1+1)+5^2(1+0)+5(1+1)</td>
<td>1535</td>
</tr>
<tr>
<td>1110</td>
<td>5^4(1+1)+5^3(1+1)+5^2(1+1)+5(1+0)</td>
<td>1555</td>
</tr>
<tr>
<td>1111</td>
<td>5^4(1+1)+5^3(1+1)+5^2(1+1)+5(1+1)</td>
<td>1560</td>
</tr>
</tbody>
</table>
b) Step 5.2: Digestion of Polynomial Values: To digest the polynomial values to the system for embedding so that maximum bit can embed another processing is done by the system that is digestion of polynomial values by subtracting 0 for 1 bit message value, 15 for two bit message value, 105 for three bit message value and 570 for four bit message value.

<table>
<thead>
<tr>
<th>Bit Stream</th>
<th>Polynomial</th>
<th>Polynomial Value</th>
<th>Digestion</th>
<th>Digested Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5(1+0)</td>
<td>5</td>
<td>5-0</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>5(1+1)</td>
<td>10</td>
<td>10-0</td>
<td>10</td>
</tr>
<tr>
<td>00</td>
<td>5^2(1+0)+5(1+0)</td>
<td>30</td>
<td>30-15</td>
<td>15</td>
</tr>
<tr>
<td>01</td>
<td>5^2(1+0)+5(1+1)</td>
<td>35</td>
<td>35-15</td>
<td>20</td>
</tr>
<tr>
<td>10</td>
<td>5^2(1+1)+5(1+0)</td>
<td>55</td>
<td>55-15</td>
<td>40</td>
</tr>
<tr>
<td>11</td>
<td>5^2(1+1)+5(1+1)</td>
<td>60</td>
<td>60-15</td>
<td>45</td>
</tr>
<tr>
<td>000</td>
<td>5^3(1+0)+5^2(1+0)+5(1+0)</td>
<td>155</td>
<td>155-105</td>
<td>50</td>
</tr>
<tr>
<td>001</td>
<td>5^3(1+0)+5^2(1+0)+5(1+1)</td>
<td>160</td>
<td>160-105</td>
<td>55</td>
</tr>
<tr>
<td>010</td>
<td>5^3(1+0)+5^2(1+1)+5(1+0)</td>
<td>180</td>
<td>180-105</td>
<td>75</td>
</tr>
<tr>
<td>011</td>
<td>5^3(1+0)+5^2(1+1)+5(1+1)</td>
<td>185</td>
<td>185-105</td>
<td>80</td>
</tr>
<tr>
<td>100</td>
<td>5^3(1+1)+5^2(1+0)+5(1+0)</td>
<td>280</td>
<td>280-105</td>
<td>175</td>
</tr>
<tr>
<td>101</td>
<td>5^3(1+1)+5^2(1+0)+5(1+1)</td>
<td>285</td>
<td>285-105</td>
<td>180</td>
</tr>
<tr>
<td>110</td>
<td>5^3(1+1)+5^2(1+1)+5(1+0)</td>
<td>305</td>
<td>305-105</td>
<td>200</td>
</tr>
<tr>
<td>111</td>
<td>5^3(1+1)+5^2(1+1)+5(1+1)</td>
<td>310</td>
<td>310-105</td>
<td>205</td>
</tr>
<tr>
<td>0000</td>
<td>5^4(1+0)+5^3(1+0)+5^2(1+0)+5(1+0)</td>
<td>780</td>
<td>780-570</td>
<td>210</td>
</tr>
<tr>
<td>0001</td>
<td>5^4(1+0)+5^3(1+0)+5^2(1+0)+5(1+1)</td>
<td>785</td>
<td>785-570</td>
<td>215</td>
</tr>
<tr>
<td>0010</td>
<td>5^4(1+0)+5^3(1+0)+5^2(1+1)+5(1+0)</td>
<td>805</td>
<td>805-570</td>
<td>235</td>
</tr>
<tr>
<td>0011</td>
<td>5^4(1+0)+5^3(1+0)+5^2(1+1)+5(1+1)</td>
<td>810</td>
<td>810-570</td>
<td>240</td>
</tr>
<tr>
<td>0100</td>
<td>5^4(1+0)+5^3(1+1)+5^2(1+0)+5(1+0)</td>
<td>905</td>
<td>905-570</td>
<td>335</td>
</tr>
<tr>
<td>0101</td>
<td>5^4(1+0)+5^3(1+1)+5^2(1+0)+5(1+1)</td>
<td>910</td>
<td>910-570</td>
<td>340</td>
</tr>
<tr>
<td>0110</td>
<td>5^4(1+0)+5^3(1+1)+5^2(1+1)+5(1+0)</td>
<td>930</td>
<td>930-570</td>
<td>360</td>
</tr>
<tr>
<td>0111</td>
<td>5^4(1+0)+5^3(1+1)+5^2(1+1)+5(1+1)</td>
<td>935</td>
<td>935-570</td>
<td>365</td>
</tr>
<tr>
<td>1000</td>
<td>5^4(1+1)+5^3(1+0)+5^2(1+0)+5(1+0)</td>
<td>1405</td>
<td>1405-570</td>
<td>835</td>
</tr>
<tr>
<td>1001</td>
<td>5^4(1+1)+5^3(1+0)+5^2(1+1)+5(1+0)</td>
<td>1410</td>
<td>1410-570</td>
<td>840</td>
</tr>
<tr>
<td>1010</td>
<td>5^4(1+1)+5^3(1+1)+5^2(1+0)+5(1+0)</td>
<td>1430</td>
<td>1430-570</td>
<td>860</td>
</tr>
<tr>
<td>1011</td>
<td>5^4(1+1)+5^3(1+1)+5^2(1+1)+5(1+0)</td>
<td>1435</td>
<td>1435-570</td>
<td>865</td>
</tr>
<tr>
<td>1100</td>
<td>5^4(1+1)+5^3(1+1)+5^2(1+1)+5(1+1)</td>
<td>1530</td>
<td>1530-570</td>
<td>865</td>
</tr>
<tr>
<td>1101</td>
<td>5^4(1+1)+5^3(1+1)+5^2(1+1)+5(1+1)</td>
<td>1535</td>
<td>1535-570</td>
<td>960</td>
</tr>
<tr>
<td>1110</td>
<td>5^4(1+1)+5^3(1+1)+5^2(1+1)+5(1+1)</td>
<td>1555</td>
<td>1555-570</td>
<td>985</td>
</tr>
<tr>
<td>1111</td>
<td>5^4(1+1)+5^3(1+1)+5^2(1+1)+5(1+1)</td>
<td>1560</td>
<td>1560-570</td>
<td>990</td>
</tr>
</tbody>
</table>

Thus for different k values the polynomial generate different class values for different length bit stream. For x=5 and k=1 different class values are given below
Here the class values will vary for different value of k. Thus the system embed the bit stream according to every pixels embedding space $\delta_{ij}$ and if there is not enough embedding space $\delta_{ij}$ then the system go for next pixel of the plane and if data is fully embedded in that plane then the system perform embedding in next plane.

C. Final Stego Image Creation

Final stego image is created by merging Spatial Domain Stego image and Transform Domain Stego image and incorporating the border cells achieved at Step 1.

D. Feature Extraction from retina Image

Retinal Recognition of a person is done by acquiring an internal body image, the retina of a person. Unlike other biometric technologies retinal recognition is not widely deployed in commercial applications now while considered invasive and expensive, retinal recognition is still the most reliable and stable means of biometric identification. Although the advantages of retinal recognition currently outweigh the disadvantages, its widespread use is held back by public acceptance. The retina is a thin layer of cells at the back of the eye ball of vertebrates. It is the part of the eye which converts light into nervous signals. The retina consists of multiple layers of sensory issue and millions of photoreceptor (cells) whose function is to transform light rays into neural impulses. These impulses subsequently travel to the brain via the optic nerve, where they are converted to images. Two distinct types of photoreceptors exist within the retina: the rods and the cones. While the cones (6 million per eye) help us to see different colors, the rods (125 million per eye) facilitate night and peripheral vision. It is the unique structure of the blood vessel pattern in the retina that forms the foundation for retinal recognition and has been used for biometric identification.

Structure of rods and cones: Rods sense to brain brightness, Cones sense color. The retina, in the back of the eye, has cells that are sensitive to light. They connect directly to your brain.

IV. MATHEMATICAL REPRESENTATION

A. Problem Formulation

The mathematical representation of the proposed method is described on embedding side and extraction side.

Let, I be a M X N matrix with each cell value $Cell_{ij}$.

Also, let us consider two series values $\alpha_{ij}$ and $\beta_{ij}$

Such that $\alpha_{ij}=1^2+2^2+3^2+4^2+5^2+6^2+\ldots$ < $Cell_{ij}$, and $\beta_{ij}=1^2+2^2+3^2+4^2+5^2+6^2+7^2+\ldots$ > $Cell_{ij}$

Then mid value, $\gamma_{ij}=(\alpha_{ij}+\beta_{ij})/2$

if $Cell_{ij}$ < $\gamma_{ij}$ then $\delta_{ij}=(Cell_{ij} - \alpha_{ij})$ else $\delta_{ij}=(\beta_{ij}-Cell_{ij})$

Let us suppose that embedding value is $Emb_{ij}$

Such that $Emb_{ij}$ < $\delta_{ij}$ and Stego Value is $S_{ij}$

if $Cell_{ij}$ < $\gamma_{ij}$ then $S_{ij}=(\alpha_{ij}+Emb_{ij})$

else $S_{ij}=(\beta_{ij}-Emb_{ij})$
The embedding polynomial used here is as follows:

\[ \sum^n = x^n (k + \text{bit}_1) + x^{n-1} (k + \text{bit}_2) + x^{n-2} (k + \text{bit}_3) + \ldots + x (k + \text{bit}_n) \]

Where, bit\_n is \text{n}^{th} bit of remainder bit stream and k= value which classify the polynomial value for different length bit stream.

On Extraction side, for each cell S\_ij of matrix I:

\[ \alpha_i := 1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2 + \ldots < \text{Cell}_ij \]
\[ \beta_i := 1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2 + 7^2 + \ldots > \text{Cell}_ij \]
\[ \gamma_i := (\alpha_i + \beta_i)/2 \]

if \ S_\_ij <= \gamma_i then \delta_i := (S_\_ij - \alpha_i) else \delta_i := (\beta_i - S_\_ij)

B. Proof of the Problem

Prove that the above formulation gives exact result on extraction side as it was embed on embedding side.

Proof. Let, I be a M X N matrix with each cell value \text{Cell}_ij.

On Embedding Side

Let us consider two series values \alpha_i and \beta_i such that \alpha_i=1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2 + \ldots < \text{Cell}_ij and \beta_i=1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2 + 7^2 + \ldots > \text{Cell}_ij

Then mid value, \gamma_i := (\alpha_i + \beta_i)/2  \hspace{1cm} (1)

if \ Cell_i <= \gamma_i then \delta_i := (\text{Cell}_ij - \alpha_i) \hspace{1cm} (2)
else \delta_i := (\beta_i - \text{Cell}_ij) \hspace{1cm} (3)

Let us suppose that embedding value is \text{Emb}_j

Such that \text{Emb}_j < \delta_j and Stego Value is \text{S}_j

if \ Cell_j <= \gamma_j then \text{S}_j := (\alpha_j + \text{Emb}_j) \hspace{1cm} (4)
else \text{S}_j := (\beta_j - \text{Emb}_j) \hspace{1cm} (5)

On Extraction Side

For each cell \text{Cell}_ij of matrix I:

\[ \alpha_i := 1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2 + \ldots < \text{Cell}_ij \]
\[ \beta_i := 1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2 + 7^2 + \ldots > \text{Cell}_ij \]
\[ \gamma_i := (\alpha_i + \beta_i)/2 \]

if \ S_\_ij <= \gamma_i then \delta_i := (S_\_ij - \alpha_i) \hspace{1cm} (6)
else \delta_i := (\beta_i - S_\_ij) \hspace{1cm} (7)

Now, using equation (4) & (5) in (6) & (7) we have

if \ S_\_ij <= \gamma_i then \delta_i := (\alpha_i + \text{Emb}_j) - \alpha_i
= \text{Emb}_j \hspace{1cm} (8)
else \delta_i := (\beta_i - \text{Emb}_j)
= \text{Emb}_j \hspace{1cm} (9)

Since we embed the value in \delta_i on embedding side and from equation (8) & (9) we get \delta_i = \text{Emb}_j Hence it is proved.

V. ALGORITHMS

In this section various algorithm for data hiding in transform domain and biometric features embedding for authentication checking has been described.

A. Pixel Selection Algorithm by Skin Tone Detection

1) Read : cover image
2) Convert : cover image \rightarrow HSV colour space
3) Get the Hue and Saturation Image Plane
4) Threshold the hue image plane and saturation image plane by setting the threshold hue_range = [0,0.11] and sat_range = [0.2,0.7] respectively.
5) Make AND operation between hue thresholded image and saturation thresholded image to get the skin tone detected image.
6) Make Dilation on the resulting image and then Erosion for noise removing.
7) Get the resulting skin tone detected image with white pixel at skin tone detected area otherwise black pixel at non skin tone detected area.

B. Algorithm for Feature Authentication for authenticity checking
1) Pixel Selection Algorithm (Spatial Domain Embedding): The pixel selection algorithm for spatial domain embedding constitutes of some algorithm, these are-
For an Image of size m x n
I= Read the image
2) Algorithm for Cropping
a) for i=2:m-1
b) for j=2:n-1
c) crpl(i-1,j-1)=I(i,j);
d) end j
e) end i
  crpl = Cropped Image
3) Algorithm for block processing
   Size of cropped image crpl is m_1 X n_1 where m_1=m-2 and n_1=n-2
   a) Initialize fi=0,fj=0
   b) for i=1: floor (m/m1/(4+1))
c) for j=1: floor (n/n1/(4+1))
d) end j
f) fi=fi+(4+1);
e) end i
4) Algorithm for Embedding: At first convert the message image into a bit stream for embedding.
   a) βij:=Add the squared value of integers up to the sum value which is below next of Pixij value
   b) αij:=Add the squared value of integers up to the sum value which is next above of Pixij value
   c) γij:=( αij+ βij)/2;
   d) Calculate the value for bit stream of length n= 4,3,2 and 1
   e) if Σ^{2}<= δij then μ_{ij}:= Σ^{i}
   else Σ^{2}<= δij then μ_{ij}:= Σ^{i}
   else Σ^{2}<= δij then μ_{ij}:= Σ^{i}
   f) if Pixij<=γij then S_{ij}:=αij+μ_{ij} else S_{ij}:=βij-μ_{ij}
   g) end j
   h) end i
   i) Stego image S generates

C. Feature Extraction from Retinal Image
Every retinal image constitutes of different type of cells which has been used in the system for authentication and those information are embedded through the transform domain.
1) Algorithm for Feature Extraction from Retinal Image
For an M X N Retinal image th
a) th1= thin image of image retina by binary image
b) Initialize N=3
c) n=(N-1)/2
d) for x:=(n+1+10):(s(1)+n-10)
e) for y:=(n+1+10):(s(2)+n-10)
set e=1
f) for k:=x-n:x+n
   set f=1
   for l:=y-n:y+n
      mat(e,f) = th1(k,l)
      f=f+1
   end l
   e=e+1
   end k
j) if(mat(2,2)==0)
k) rods(x,y) = sum value of occurrence of zeros to mat
l) cons(x,y) = sum value of occurrence of zeros to mat
m) end y
n) end x
o) find rods and cons cell values

D. Proposed Method Algorithm for Data Hiding in Transform Domain

1) Pixel Selection Algorithm for Transform Domain Embedding: The pixel selection algorithm for Transform domain embedding constitutes of some algorithm, these are for an Image of size m x n.

Step1: I= Read the image
Algorithm for Cropping
for i=2:m-1
   for j=2:n-1
      crpI(i-1,j-1)=I(i,j);
   end j
end i
   crpI = Cropped Image

2) Algorithm for Block Processing using DCT
Size of cropped image crpI is m1 X n1 where m1=m-2 and n1=n-2
a) Initialize fi=0,fj=0,fqi=0,fqj=0;
b) for i=1: floor (m1/(4+1))
c) for j=1: floor (n1/(4+1))
d) for i=1:4
   e) for j=1:4
      blk=crpI((ii+1)+fi,(jj+1)+fj);
   end j
end i
f) end jj
g) end ii
h) Dblk=dct2conversion of blk;
i) for i=1:4
   j) for j=1:4
      DcrpI((ii+1)+fqi,(jj+1)+fqj)=Dblk(ii,jj);
   end j
end i
m) fj=fj+(4+1);
n) fqj=fqj+(4+1);
o) end j
p) fi=fi+(4+1);
q) fj=0;
r) fqj=fqj+(4+1);
s) fqj=0
t) end i

Let DrpI = DCT processed image and rDrpl = ignoring the element of top left corner of each block of Drpl.

3) Algorithm for integer value, sign and remainder extraction from Drpl: Size of DCT processed image is m2 x n2
   a) for i=1:m2
   b) for j=1:n2
   c) if Drpl(I,j)>0
      VDrpl(i,j)=floor(rDrpl(i,j));
      Sign(i,j)=1;
      rem(i,j)= rDrpl(I,j)-floor(rDrpl(i,j));
   d) else
      VDrpl(i,j)=floor(-rDrpl(i,j));
      Sign(i,j)=-1;
      rem(I,j)=(-rDrpl(i,j))-floor(-rDrpl(i,j));
   e) end if
   f) end j
   i) end i

Integer value matrix is VDrpl. Sign matrix is Sign. Remainder matrix is rem.

5) Algorithm for Embedding: At first convert the message image into a bit stream for embedding.

For an m2 x n2 matrix VDrpl for i=1:m2
   for j=1:n2
   For each cell Cellij in the matrix VDrpl:
   a) αij:=Add the squared value of integers up to the sum value which is below next of Cellij value (for example, Cellij=127
      Then, αij=1^2+2^2+3^2+4^2+5^2+6^2=91< 127)
   b) βij:=Add the squared value of integers up to the sum value which is next above of Cellij value (for example, Cellij=127
      βij=1^2+2^2+3^2+4^2+5^2+6^2+7^2=140 >127)
   c) γij=( αij+ βij)/2;
   d) if Cellij<= γij then δij=( Cellij - αij) else δij=( βij- Cellij)
   e) Calculate the value for bit stream of length n= 4,3,2 and 1
      ∑n:=x^n.(k+bit1)+λx^n-1.(k+bit2)+x^n-2.(k+bit3)+.............+x.(k+bitn)
   f) if ∑n<= δij then μij:= ∑n
      a. else if ∑n<= δij then μij:= ∑n
      b. else if ∑n<= δij then μij:= ∑n
      c. else if ∑n<= δij then μij:= ∑n
   g) if Cellij<= γij then Sij:= αij+ μij else Sij:= βij - μij
   h) end j
   i) end i

6) Algorithm for Stego Image Creation in Transform Domain

Algorithm for Sign Restoration
   a) for i=1:m2
   b) for j=1:n2
   c) V(i,j)=sign(i,j) * (S(i,j)+rem(i,j))
   d) end j
   e) end i

7) Algorithm for Top Most Left Corner Value Restoration
   a) for i=1:m2
   b) for j=1:n2
   c) V1(i,j)=mask11(i,j)+V(i,j);
   d) end j
   e) end i
mask11 hold the only value of top most left corner of each block

8) Algorithm for Block Processing using Inverse DCT
   a) Initialize fi=0,fj=0,fqi=0,fqj=0;
   i) for i=1: floor (m/(4+1))
   ii) for j=1: floor (n/(4+1))
   iii) for ii=1:4
   iv) for jj=1:4
        blk=V1((ii+1)+fi,(jj+1)+fj);
   v) end jj
   vi) end ii
   vii) iDblk= inverse dct2conversion of blk
   viii) for ii=1:4
   ix) for jj=1:4
        iDcrpI((ii+1)+fqi,(jj+1)+fqj)=iDblk(ii,jj)
   x) end jj
   xi) end ii
   xii) fj=fj+(4+1)
   xiii) fqi=fqi+(4+1)
   xiv) end j
   xv) for jj=1:4
        iDcrpI((ii+1)+fqi,(jj+1)+fqj)=iDblk(ii,jj)
   xvi) fi=fi+(4+1)
   xvii) fj=0;
   xviii) fqi=0;
   xix) end i

9) Algorithm for Ignored Cell Value Assignment
   a) Initialize fi=0,fj=0
   b) for i=1: floor (m/(4+1))
   c) for j=1: floor (n/(4+1))
   d) for ii=1:4
   e) for jj=1:4
        I((ii+1)+fi,(jj+1)+fj)=iDcrpI(I,j);
   f) end jj
   g) end ii
   h) fj=fj+(4+1)
   i) end j
   j) fi=fi+(4+1)
   k) fj=0;
   l) end i
   m) Stego=I
   n) Stego image is Stego.

10) Algorithm for Extraction
    For an M X N Stego Image
        for i=1:M
        for j=1:N
            For each cell Cellij in Original image:-
            a) αij:=Add the squared value of integers up to the sum value which is below next of Cellij value
            b) βij:=Add the squared value of integers up to the sum value which is next above Cellij value
            c) γij:( αij+ βij)/2
            d) if Sij<= γij then δij:=( Sij - αij) else δij:=( βij - Sij)
VI. ANALYSIS OF THE RESULT

In this section the experimental results of the proposed method has been described. Table IV below shows the extracted features of different thumb image used for authentication as well as extracted secret message.

<table>
<thead>
<tr>
<th>Authentication Image</th>
<th>Message Retinal Image</th>
<th>Resulting Extracted Feature Message of Retinal Image from Spatial Domain</th>
<th>Resulting Extracted Secret Message of Retinal Image from Frequency Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retina1.jpg</td>
<td>RodCount=5 ConeCount=50</td>
<td>Message=AAAAAAAAAAAAAAAAAAAA</td>
<td>Message=AAAAAAAAAAAAAAAAAAAA</td>
</tr>
</tbody>
</table>

Fig 13. GUI of Feature Extraction from retina Image
The experimental results of proposed method have been evaluated based on two benchmarks criteria. First one is the capacity of hidden data and the second one is the imperceptibility or the quality of the stego image. This method works in a combined approach of information hiding as well as authenticity checking. The capacity of hidden data depends upon the bit stream to be embed and embedding space which depends upon the pixel value so that capacity varies from message to message. Imperceptibility or the quality of stego image produced by this method has been tested thoroughly based on various image similarity metrics namely MSE, PSNR, CORELATION, RMSE, SSIM, KLDIV and ENTROPY. Fig. 14 to 20 shows the graphical representation of calculated value of various similarity metrics for images.

A. Mean Square Error (MSE)

It is computed by averaging the squared intensity of the cover and stego image pixels. The “(10)” shows the MSE [39].

\[
MSE = \frac{1}{NM} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} e(m,n)^2
\]

Where NM is the image size \((N \times M)\) and \(e(m,n)\) is the reconstructed image.
B. Peak Signal-to-Noise Ratio (PSNR)

A mathematical measure of image quality is Signal-to-noise ratio (SNR), which is based on the pixel difference between two images. PSNR [39] shows in equation“(11)”. Where, S stands for maximum possible pixel value of the image. If the PSNR is greater than 36 DB then the visibility looks same between cover and stego image, so HVS not identified the changes.

\[
PSNR = 10 \log_{10} \frac{S^2}{MSE} \quad \text{(11)}
\]

C. Correlations

Pearson’s correlation coefficient [40] is widely used in statistical analysis as well as image processing. Here apply it in Cover and Stego images to see the difference between these two images. The Correlation shows in equation“(12)”.

\[
r = \frac{\sum_{i=1}^{n} (x_i - \overline{x}) \cdot (y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \overline{x})^2 \cdot \sum_{i=1}^{n} (y_i - \overline{y})^2}} \quad \text{(12)}
\]

The \(X_i\) and \(Y_i\) are the cover image and bar of \(X\) and \(Y\) are stego image positions. The correlation values are tens to 1 that means both the images are likely to same. RMSE [41] is one kind of measurement of difference between values of Cover Image and the values of Stego Image. The RMSE shows in equation“(13)”. 

\[
RMSE = \sqrt{\frac{\sum_{i=1}^{n} (X_{\text{obs},i} - X_{\text{sto},i})^2}{n}} \quad \text{(13)}
\]
X_{obs,i} and X_{mod,i} are two image vectors i.e. cover and stego.

D. Structural Similarity Index (SSIM)
Wang et al. [42], proposed Structural Similarity Index concept between original and distorted image. The Stego and Cover images are converted into vectors. Then two means and two standard derivations and one covariance value are computed. Then the SSIM [42] computed between Cover and Stego images. SSIM shows in equation “(14)”.

\[ SSIM = \frac{(2\mu_x \mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\mu_x^2 + \mu_y^2 + C_2)} \] ....(14)

It has been observed that the SSIM values are nearest to 1, which shows that the cover and the stego both are prone to parallel and our human visual system can’t recognize the changes occurred in the images.

E. K L Divergence (KLDIVG)
With the help of probability density function (PDF) for each Image (cover and stego) we estimate the Kullback-Leibler Divergence [43]. KL divergence shows in equation “(15)”.
Entropy is a measure of the uncertainty associated with a random variable [44]. Here, a 'message' means a specific realization of the random variable. The equation “(16)” shows it.

\[ \Delta S = \int \frac{dQ_{\text{rev}}}{T} \]  

Where, \( S \) is the entropy; \( T \) is the uniform thermodynamic temperature of a closed system divided into an incremental reversible transfer of heat into that system (dQ).
Here we have tested through some steganalysis technique because to access the security of the Steganography algorithm the attack is necessary. First exact detector of LSB replacement was the heuristic RS analysis [45]. Then Sample Pairs (SP) analysis was analyzed and reformulated by Dumitrescu et al. [46] in 2002. In this work all the stego image is generated by the help of our algorithm and tested through steganalysis attack algorithm i.e. RS analysis. Fig. 21 shows the Analysis of attack of an RGB image as cover and stego image.

VII. CONCLUSION

In this paper a new and efficient approach of retinal biometric steganography technique has been proposed which works as a combination of spatial domain and transforms domain image steganography. This technique has been designed to incorporate both hiding and authentication aspects. Hiding of secret message has been done in frequency domain whereas authentication principle with the help of retina has been incorporated in spatial domain. In almost all the existing methods hiding of information has been done only either in spatial domain or in transform domain. In this work a humble attempt has been made to integrate the concept of information hiding and authentication principle with the aid of biometric security.

REFERENCES


[31] Shangling Song; Kazuhiko Ohnuma; Zhi Liu; Liangmo Mei; Akira Kawada; Tomoyuki Monma "Novel biometrics based on nose pore recognition" 2009.


[34] Probing the uniqueness and randomness of IrisCodes: Results from 200 billion iris pair comparisons.” Proceedings of the IEEE, vol. 94 (11), 2006, pp. 1927-1935.


