CHAPTER 7

GEOMETRIC ROBUST MULTIMODAL BIOMETRIC WATERMARKING SCHEME FOR COPYRIGHT PROTECTION OF DIGITAL IMAGES

7.1 Introduction

This chapter proposes a copyright protection system using multimodal biometrics for content protection of digital images. Two biometric traits, one physiological (face) and one behavioral (offline handwritten signature) of the legitimate owner are used as watermarks. These watermarks are embedded at different levels of resolution to the high frequency wavelet coefficients through optimal control of the embedding factor. Experimental evaluation confirms the imperceptibility and robustness of the scheme against a family of typical image distortions functions like JPEG Compression, filtering etc as well as geometric attacks such as scaling, rotation and combined attacks.

7.2 Proposed Method

In the proposed algorithm, the classic Cox’s digital watermarking [104] algorithm concept has been suitably modified to embed two biometric traits i.e. an offline handwritten signature and a face image of the owner of digital image. The host image is decomposed using lifting wavelet transform using biorthogonal wavelets. The watermarks are embedded at different levels of resolution hence providing resilience to the proposed algorithm. The biometric images used are gray scale images and which are binarized using algorithm described in [88].

7.2.1 Watermark Embedding

The following steps are followed for the watermark embedding.

- Let \( I_{original} \) be the host image of size \( M \times N \).

- Let \( w_{m_1} \) and \( w_{m_2} \) be the signature watermark and face watermark of length \( (M_s \times N_s) \) and \( (M_f \times N_f) \) respectively. The feature vectors are generated after binarization by converting the watermarks into a 1D array of size \( ((M_s \ast N_s), 1) \) and \( ((M_f \ast N_f), 1) \) resp.
• Secret key $K1$ is entered to generate pseudorandom sequences for signature watermark insertion.

• For embedding the signature image, the Lifting Wavelet Transform $I_{lwt(i,j)}$ of the host image is calculated for level 2 decomposition, sub-bands of size $\frac{N}{2^2} \times \frac{N}{2^2}$ are obtained.

• The horizontal detailed band is used for watermark insertion as embedding in the approximation band causes more perceptible changes to the watermarked image. The watermark is embedded as follows:

\[
\text{for } i = 1: \text{length}(wm_1)
\]

\[
\text{random\_seq\_1} = \text{round}(2 \ast (\text{rand}(M/4, N/4) - 0.5));
\]

\[
\text{if } (wm_1(i) == 0)
\]

\[
HH_2 = HH_2 + k \ast \text{random\_seq\_1};
\]

\[
\text{end}
\]

\[
\text{end}
\]

$k$ is the strength factor used for embedding. Any visible effect, such as blocking, due to manipulation of these wavelet coefficients is controlled using parameter optimization method as described in the subsequent section. Experimentally it has been observed that this factor must be $<1$.

• Secret key $K2$ is entered to generate pseudorandom sequences for face watermark insertion.

• For embedding the face image, the Lifting Wavelet Transform $I_{lwt(i,j)}$ of the host image is calculated for level 3 decomposition, sub-bands of size $\frac{N}{2^3} \times \frac{N}{2^3}$ are obtained.

• The level 3 horizontal detailed band is used for embedding the face image as follows:

\[
\text{for } i = 1: \text{length}(wm_2)
\]
\[ \text{random_seq}_2 = \text{round}(2 \times (\text{rand}(M/8, N/8) - 0.5)) \]

\[ \text{if } (\text{wm}_2(i) == 0) \]

\[ \text{HH}_3 = \text{HH}_3 + k \times \text{random_seq}_2; \]

\[ \text{end} \]

\[ \text{end} \]

- The watermarked image \( I_{wm} \) is obtained by applying inverse LWT twice as shown in fig 7.1.
7.2.2 Watermark Extraction

- The extraction process uses the correlation detector to recover the embedded watermarks by using the same secret keys for random sequence generation as were used for embedding. Face watermark is initially extracted after the user has entered the secret key $K_2$ as follows:

\[
\text{for } (i = 1: \text{length}(wm_2))
\]

\[
\text{random_seq}_2 = \text{round}(2 \times (\text{rand}(M/8, N/8) - 0.5));
\]

\[
\text{correlation\_face}(i) = \text{corr2}(HH_3, \text{random\_seq}_2);
\]

\[\text{end}\]

\[
\text{if } (\text{correlation\_face}(i) > \text{std}(\text{correlation\_HH}_3))
\]

\[
\text{wm}_2 (i) = 0;
\]

\[
\text{else } \text{wm}_2 (i) = 1;
\]

\[\text{end}\]

- The extraction process uses the correlation detector to recover the embedded watermarks by using the same secret keys for random sequence generation as were used for embedding. Signature watermark is extracted after the user has entered the secret key $K_1$ as follows:

\[
\text{for } (i = 1: \text{length}(wm_1))
\]

\[
\text{random\_seq}_1 = \text{round}(2 \times (\text{rand}(M/4, N/4) - 0.5));
\]

\[
\text{correlation\_signature}(i) = \text{corr2}(HH_2, \text{random\_seq}_1);
\]

\[\text{end}\]

\[
\text{if } (\text{correlation\_signature}(i) > \text{std}(\text{correlation\_HH}_2))
\]

\[
\text{wm}_1 (i) = 0;
\]

\[
\text{else } \text{wm}_1 (i) = 1;
\]
• The same is depicted in Figure 7.2.

![Diagram showing the process of watermarking extraction.]

**Fig 7.2:** Block Diagram of Proposed Watermarking Extraction Scheme.

## 7.3 Experimental Results and Discussions

### 7.3.1 Experimental Setup

It is important to evaluate an image watermark algorithm on many different images. Images should cover a broad range of contents and types. The proposed technique has been tested on standard evaluation images for watermarking algorithms. The standard Yale database for 50 face images has been used and a signature database consisting of signatures of 50 users has been used for experimental purposes. Figure 7.3 shows the original image as well as watermarked image and the original as well as extracted watermarks. In all the above images, the same face and signature image has been used as a watermark.
Fig 7.3: The Original and Recovered Face and Signature Watermark after Applying the Embedding and Extraction Algorithm on Standard Test Image Using the Same Signature and Face Image (a) Plane (b) Baboon (c) Lena (d) Barbara (e) Cameraman (f) Boat.
7.3.2 Robustness Tests under StirMark Attacks

The robustness of the proposed algorithm has been tested against the benchmark StirMark attacks. These attacks include common signal processing attacks like median filtering, Gaussian noise addition, Jpeg Compression, or geometric attacks like resizing, rotation or a combination of both.

7.3.2.1 Median Filtering

The most common signal processing attack in digital image is filtering. Table 7.1 shows the MSSIM values for the extracted face and signature watermarks, after applying median filter of varying filter lengths. It can be observed that even after the degradations caused in the images, the recovered watermarks are very much recognizable. Some results for the same are shown in Figure 7.4.
**Fig 7.4:** Extracted Signature and Face Watermarks After Median Filtering on Different Images Using Different Filter Lengths.
Table 7.1: MSSIM Results of Extracted Watermarks Median Filtering with Different Filter Sizes Attack.

<table>
<thead>
<tr>
<th>Image</th>
<th>3 x 3</th>
<th>5 x 5</th>
<th>7 x 7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Face</td>
<td>Signature</td>
<td>Face</td>
</tr>
<tr>
<td>Barbara</td>
<td>0.99748</td>
<td>0.99568</td>
<td>0.99746</td>
</tr>
<tr>
<td>Lena</td>
<td>0.99644</td>
<td>0.99536</td>
<td>0.99644</td>
</tr>
<tr>
<td>Boat</td>
<td>0.99636</td>
<td>0.99627</td>
<td>0.99616</td>
</tr>
<tr>
<td>Plane</td>
<td>0.99742</td>
<td>0.99632</td>
<td>0.99748</td>
</tr>
<tr>
<td>Baboon</td>
<td>0.99618</td>
<td>0.99512</td>
<td>0.99616</td>
</tr>
</tbody>
</table>

7.3.2.2 Gaussian Noise

Noise is added to image to degrade its quality. Robustness against additive Gaussian noise is estimated by degrading the watermark image by randomly adding Gaussian noise with mean 0 and variance 0.01. Table 7.2 shows the MSSIM values for various recovered watermarks while Figure 7.5 shows the results pictorially.

Table 7.2: MSSIM Results of Extracted Watermarks under Gaussian Filtering Attack Mean=0, Variance=0.01.

<table>
<thead>
<tr>
<th>Image</th>
<th>Gaussian filtering</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Face</td>
</tr>
<tr>
<td>Barbara</td>
<td>0.99671</td>
</tr>
<tr>
<td>Lena</td>
<td>0.99753</td>
</tr>
<tr>
<td>Boat</td>
<td>0.99597</td>
</tr>
<tr>
<td>Plane</td>
<td>0.99753</td>
</tr>
<tr>
<td>Baboon</td>
<td>0.99747</td>
</tr>
</tbody>
</table>
Fig 7.5: Extracted Watermarks after Addition of Gaussian Noise to Watermarked Image.
7.3.2.3 Scaling

The images were either cropped or resized. Table 3.3 and fig 3.8 present the results of the effect of various scaling factors on the watermarked images.

Table 7.3: MSSIM Results of Extracted Watermarks Under Scaling Attack.

<table>
<thead>
<tr>
<th>Image</th>
<th>0.5</th>
<th>0.75</th>
<th>1.5</th>
<th>2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Face</td>
<td>Signature</td>
<td>Face</td>
<td>Signature</td>
</tr>
<tr>
<td>Barbara</td>
<td>0.99682</td>
<td>0.99485</td>
<td>0.99648</td>
<td>0.99552</td>
</tr>
<tr>
<td>Lena</td>
<td>0.99597</td>
<td>0.99539</td>
<td>0.99629</td>
<td>0.99619</td>
</tr>
<tr>
<td>Boat</td>
<td>0.99735</td>
<td>0.99523</td>
<td>0.99723</td>
<td>0.99741</td>
</tr>
<tr>
<td>Plane</td>
<td>0.99665</td>
<td>0.99526</td>
<td>0.99674</td>
<td>0.9966</td>
</tr>
<tr>
<td>Baboon</td>
<td>0.99562</td>
<td>0.99643</td>
<td>0.99595</td>
<td>0.99587</td>
</tr>
</tbody>
</table>

(a)
Fig 7.6: Effect Of Various Scaling Factors On The Recovered Watermarks (a) 0.5 (b) 0.75 (c) 1.5 (d) 2.0.

7.3.2.4 JPEG Compression

Another most common manipulation in digital image is image compression. To check the robustness against Image Compression, the watermarked image is compressed using JPEG compression for different compression ratios. MSSIM values for the extracted watermarks are depicted in Table 7.4 and subsequently in Figure 7.7.
Table 7.4: MSSIM Results for JPEG Compression with Varying Q-factor.

<table>
<thead>
<tr>
<th>Image</th>
<th>20</th>
<th>40</th>
<th>60</th>
<th>80</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Face</td>
<td>Sign</td>
<td>Face</td>
<td>Sign</td>
<td>Face</td>
</tr>
<tr>
<td>Barbara</td>
<td>0.99593</td>
<td>0.99552</td>
<td>0.999596</td>
<td>0.99568</td>
<td>0.99587</td>
</tr>
<tr>
<td>Lena</td>
<td>0.99739</td>
<td>0.99614</td>
<td>0.99743</td>
<td>0.99592</td>
<td>0.99738</td>
</tr>
<tr>
<td>Boat</td>
<td>0.99743</td>
<td>0.99695</td>
<td>0.9976</td>
<td>0.99684</td>
<td>0.9976</td>
</tr>
<tr>
<td>Plane</td>
<td>0.9972</td>
<td>0.99476</td>
<td>0.99714</td>
<td>0.99495</td>
<td>0.99714</td>
</tr>
<tr>
<td>Baboon</td>
<td>0.99631</td>
<td>0.99545</td>
<td>0.99621</td>
<td>0.99521</td>
<td>0.99624</td>
</tr>
</tbody>
</table>

(a)
Fig 7.7: Extracted watermarks after JPEG Compression (a) Q-factor = 20 (b) Q-factor = 40
(c) Q-factor = 60 (d) Q-factor = 80 (e) Q-factor = 100
7.3.2.5 Rotation

In the next experiment, the robustness of the proposed algorithm is tested against rotation attack. The proposed embedding approach is robust to rotation provided that one can compensate for the loss of synchronization. Table 7.5 and Figure 7.8 shows the MSSIM values after the watermarked image was rotated in the range of (-5 degree to 5 degree).

**Table 7.5:** MSSIM Results of Extracted Watermarks Under Rotation Attack.

<table>
<thead>
<tr>
<th>Image</th>
<th>Rotation Angle (Degrees)</th>
<th>-5</th>
<th>-1</th>
<th>1</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Face</td>
<td>Signature</td>
<td>Face</td>
<td>Signature</td>
<td>Face</td>
</tr>
<tr>
<td>Barbara</td>
<td>0.99744</td>
<td>0.99512</td>
<td>0.9974</td>
<td>0.99501</td>
<td>0.99719</td>
</tr>
<tr>
<td>Lena</td>
<td>0.99659</td>
<td>0.99454</td>
<td>0.99613</td>
<td>0.99456</td>
<td>0.99641</td>
</tr>
<tr>
<td>Boat</td>
<td>0.99669</td>
<td>0.99522</td>
<td>0.99661</td>
<td>0.99566</td>
<td>0.9964</td>
</tr>
<tr>
<td>Plane</td>
<td>0.99705</td>
<td>0.99524</td>
<td>0.99711</td>
<td>0.99537</td>
<td>0.99697</td>
</tr>
<tr>
<td>Baboon</td>
<td>0.99729</td>
<td>0.99553</td>
<td>0.99705</td>
<td>0.99539</td>
<td>0.99721</td>
</tr>
</tbody>
</table>

(a)
Fig 7.8: Extracted Watermarks after (a) -5 Degree Rotation (b) 5 Degrees Rotation (c) -1 Degree Rotation
(d) 1 Degree Rotation.

7.3.2.6 Combined Attacks

In addition to the common individual attacks, the robustness of the algorithm was also tested against the combined attacks. The results of the same are presented in table 7.6.
Table 7.6: MSSIM Results of Extracted Watermarks Under Combined Attacks.

<table>
<thead>
<tr>
<th>Attack</th>
<th>Baboon</th>
<th>Barbara</th>
<th>Boat</th>
<th>Lena</th>
<th>Plane</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Face</td>
<td>Sign</td>
<td>Face</td>
<td>Sign</td>
<td>Face</td>
</tr>
<tr>
<td>JPEG_20 + Scaling(0.5)</td>
<td>0.99584</td>
<td>0.99669</td>
<td>0.99682</td>
<td>0.99539</td>
<td>0.99618</td>
</tr>
<tr>
<td>JPEG_20 + Scaling(2.0)</td>
<td>0.99583</td>
<td>0.99723</td>
<td>0.99697</td>
<td>0.99503</td>
<td>0.99759</td>
</tr>
<tr>
<td>JPEG_20 + Med_filt (3x3)</td>
<td>0.99618</td>
<td>0.99512</td>
<td>0.99748</td>
<td>0.99568</td>
<td>0.99536</td>
</tr>
<tr>
<td>JPEG_20 + Med_filt</td>
<td>0.99616</td>
<td>0.99509</td>
<td>0.99746</td>
<td>0.99609</td>
<td>0.99528</td>
</tr>
<tr>
<td>JPEG_20 + Med_filt (5x5)</td>
<td>0.99604</td>
<td>0.99511</td>
<td>0.99749</td>
<td>0.99583</td>
<td>0.99627</td>
</tr>
<tr>
<td>JPEG_20 + Med_filt (7x7)</td>
<td>0.99769</td>
<td>0.99487</td>
<td>0.99644</td>
<td>0.99581</td>
<td>0.99613</td>
</tr>
</tbody>
</table>

7.4 Conclusion

This chapter presented a novel biometric multimodal watermarking algorithm using LWT. The watermark is embedded into the high frequency areas of the cover image using Watson’s HVS criterion. Watermark detection is done using the correlation detector with the help of biorthogonal wavelets to make the recovery of the watermark better. A delicate balance has been maintained in the trade-off between imperceptibility and robustness through the multi-objective optimization approach. The robustness of the proposed method has been experimentally validated using the standard test benchmarks and the results are far more superior than any approach used for multimodal biometric embedding. The work can be further extended by incorporating pattern algorithms for the purpose of authentication of the extracted watermark.