CHAPTER-5
WATERMARKING OF COLOR IMAGES

5.1 INTRODUCTION

After satisfactorily developing the watermarking schemes for gray level images, we focused on developing the watermarking schemes for the color images. The proposed ICAR watermarking scheme given in the previous chapter was chosen as a base as it has already proved its resistance to JPEG compression attack and other common image manipulations and performed at par with other state-of-the-art watermarking schemes. In this chapter, we conducted a study to find out the suitability of color channel (Red/Green/Blue) to hide the watermark data while using the DCT based watermarking scheme. We present an ICAR watermarking schemes for true colored BMP images.

5.2 PERFORMANCE ANALYSIS OF COLOR CHANNEL FOR DCT BASED IMAGE WATERMARKING SCHEME

Initially, the suitability of color channel to hide a monochromatic watermark in a 24-bit color Window’s BMP image while using classical MBCE watermarking scheme, was examined as MBCE scheme is the base scheme used to develop the proposed ICAR watermarking schemes.

Four well known 24 bit colored test images of Lena, Pepper, Mandrill and Monarch (Size 512 x 512 pixels), shown in Figure 3.12 were taken. The watermark logo used is shown in Figure 3.13. Then, the watermark logo was embedded in these 4 test images using the MBCE watermarking scheme. To analyze the performance of Red, Green and Blue channels, the watermark was embedded separately in R, G and B channels one by one. So, total 4 images were watermarked and each of these images were watermarked thrice.
Table 5.1: PSNR of Extracted watermark from JPEG compressed watermark test images

<table>
<thead>
<tr>
<th></th>
<th>LENABMP</th>
<th></th>
<th>PEPPER.BMP</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>JPEG Compression</strong></td>
<td><strong>Q = 20</strong></td>
<td><strong>Q = 40</strong></td>
<td><strong>Q = 60</strong></td>
<td><strong>Q = 80</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MANDRILL.BMP</strong></td>
<td><strong>Q = 20</strong></td>
<td><strong>Q = 40</strong></td>
<td><strong>Q = 60</strong></td>
<td><strong>Q = 80</strong></td>
</tr>
<tr>
<td><strong>BLUE</strong></td>
<td>4.00899</td>
<td>4.72608</td>
<td>11.0469</td>
<td>17.8341</td>
</tr>
</tbody>
</table>

**Figure 5.1:** Recovered watermarks for Lena.bmp after jpeg attack at Q = 40

After watermarking the test images in all three color channels, we compressed all 12 watermarked images using JPEG compression at 4 JPEG quality factors (Q = 80, 60, 40, and 20) and then recovered the watermark logos from JPEG compressed images. We calculated the PSNR values of all these 12 x 4 = 48 extracted watermark logos. Table 5.1 summarizes their PSNR values. The recovered watermark logos from all 3 Lena’s test images, if they were JPEG compressed at Q = 40, are shown in Figure 5.1. It was observed that for all test images, quality of extracted watermark logo was better, if watermark is embedded in GREEN channel for all JPEG quality factors. This can be justified as follows:

JPEG uses the YCbCr color model. While converting from BMP to JPEG, following color transformation occurs:

\[
Y' = 0.299 \times R + 0.587 \times G + 0.114 \times B
\]

\[
Cb = 128 - 0.168 \times R - 0.331 \times G + 0.500 \times B
\]

\[
Cr = 128 + 0.500 \times R - 0.419 \times G - 0.081 \times B
\]

--------- (5.1)
Where $Y'$ is the luminance component and $Cb$ and $Cr$ are the blue and red chrominance components. $Y'CbCr$ is not an absolute color space. It is a way of encoding RGB information and the actual color displayed depends on the actual RGB colorants used to display the signal. It is clear from Equations 5.1 that $G$ is multiplied by relatively larger coefficient and thus green channel should carry the watermark data for the better recovery if images are JPEG compressed after the watermarking using the MBCE scheme.

Now to further validate the concept of ‘preprocessing’ introduced in previous chapter, color channels of all test images were histogram equalized one at a time, i.e., Lena image had now 3 copies where in one copy only red channel is equalized, in another copy only green channel is equalized and in the third copy only blue channel is equalized leading to 12 test images to be watermarked. The watermark logo was embedded in the histogram equalized color channel for all 12 test images. We performed the following attacks on the watermarked images:

1) JPEG Attack (low JPEG compression with $Q = 20$);
2) Noise Attack (adding 10% Gaussian noise in the watermarked images); and
3) Histogram Equalization (equalizing the histogram of the watermarked images).

The watermark logos were recovered from the attacked images and their PSNR values were calculated. Table 5.2 summarizes the PSNR values of watermark logos recovered. It may be observed from Table 5.2 that for all cases if a color channel of the image was HISTOGRAM EQUALIZED before embedding the watermark, recovery of watermark is better i.e. PSNR values are higher. Therefore, our proposed idea of ‘preprocessing’ worked well for colored BMP images also. It may be further observed that the difference in the PSNR values of recovered logos from original image and equalized image are high in the case of “histogram equalization” attack because our preprocessing step is itself the histogram equalization. These results further prove that a modification in the image such that the effect after the attack on the watermarked image could be minimized, increases the robustness against that attack for colored images watermarking algorithm.
It is, therefore, concluded that to decide the color channel to carry the watermark data, we will have to analyze the characteristics of attack itself. If there is high probability that watermark image may undergo JPEG compression, we should select the GREEN channel because while converting to JPEG format, green channel’s data has higher impact as compared to other color channel’s data.

<table>
<thead>
<tr>
<th>Color Channel</th>
<th>Attack</th>
<th>LENABMP</th>
<th>PEPPER.BMP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Jpeg Q20</td>
<td>Jpeg Q20</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Histogram Equalization</td>
<td>Histogram Equalization</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Noise (12.5%)</td>
<td>Noise (12.5%)</td>
</tr>
<tr>
<td>RED</td>
<td>Original</td>
<td>3.85853</td>
<td>15.5867</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4.9277</td>
<td>3.89485</td>
</tr>
<tr>
<td></td>
<td>Equalized</td>
<td>4.7334</td>
<td>15.6074</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5.13784</td>
<td>4.706</td>
</tr>
<tr>
<td>GREEN</td>
<td>Original</td>
<td>6.2285</td>
<td>15.7285</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5.25512</td>
<td>4.50915</td>
</tr>
<tr>
<td></td>
<td>Equalized</td>
<td>6.8358</td>
<td>18.7032</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5.37656</td>
<td>6.9542</td>
</tr>
<tr>
<td>BLUE</td>
<td>Original</td>
<td>3.78205</td>
<td>16.8769</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4.96932</td>
<td>3.70676</td>
</tr>
<tr>
<td></td>
<td>Equalized</td>
<td>4.1447</td>
<td>23.0387</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5.41004</td>
<td>4.1985</td>
</tr>
<tr>
<td>MANDRILL.BMP</td>
<td>Original</td>
<td>4.58561</td>
<td>17.1942</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8.9886</td>
<td>4.18767</td>
</tr>
<tr>
<td></td>
<td>Equalized</td>
<td>5.2266</td>
<td>16.7228</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.36638</td>
<td>4.7169</td>
</tr>
<tr>
<td>MONARCH.BMP</td>
<td>Original</td>
<td>7.3024</td>
<td>17.4885</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.2963</td>
<td>4.88113</td>
</tr>
<tr>
<td></td>
<td>Equalized</td>
<td>11.1118</td>
<td>21.6698</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.81065</td>
<td>6.9542</td>
</tr>
</tbody>
</table>

84
It is also clear from Table 5.2 that for attacks other than JPEG Compression, performance of color channels for all images had no fixed pattern which means that robustness may depend upon the attack characteristics as well as “image characteristics” also.

Therefore, the goal for the further development was not only to develop an ICAR watermarking scheme but also to find out some relationship between the performances of our proposed schemes with the “image characteristics” itself.

5.3 **DEVISING AN ICAR WATERMARKING SCHEME FOR COLORED BMP IMAGES**

In the previous chapter, we have proposed an ICAR scheme for watermarking gray level image. Results indicated that this scheme was not only an ICAR scheme but also very robust to JPEG compression attack and other common image manipulations. Therefore, we decided to extend the same approach for colored BMP images also. In the earlier proposed ICAR scheme, we have introduced redundancy in swapping and made the swapping criterion dependent on low frequency coefficient. To further improve the robustness, we propose a new swapping criterion with the assurance that no two watermarked copies of an image have same policy of watermarking. An attacker may attack on large number of middle band coefficients but if image has to remain perceptually unchanged, the average value (Av) of all middle band coefficients (total 22 in numbers) will not modify to a great extent. So, unlike the previous scheme where we swapped 4 pairs, we swapped 4 middle band coefficients (not pair) with the “Av” value. Details of this swapping mechanism are described in Section 5.3.3.

The proposed watermarking scheme can be defined as a 7-tuple \((X, W, P, T, G, E, D)\) where:

1) \(X\) denotes the set of instances \(X_i\) of a particular gray level image, (If \(N\) copies of an image are to be watermarked, then \(0 \leq i \leq N\));

2) \(W\) denotes the monochrome watermark logo;

3) \(P\) denotes the set of policies \(P_i, 0 \leq i \leq N\);
4) “T” is the “watermark strength parameter”;

5) G denotes the policy generator algorithm $G: X_i \rightarrow P_i$, where each $X_i$ will have a unique $P_i$, i.e. a different policy to hide the watermark data.

6) E denotes the watermark embedding algorithm, $E: X_i \times W \times P_i \rightarrow X_i'$;

7) D denotes the watermark detection algorithm, $D: X_i' \times P_i \rightarrow W'$, where $W'$ represents extracted watermark.

The parameter “T” is analogous to “K” of classical MBCE scheme. In classical MBCE scheme, relative strength of 2 coefficient’s value of FM region decides the decoding of “1” or “0”. If the relative strength of 2 values has to decide the decoding of “0” or “1”, then larger value should remain larger even after image manipulations. So, we adjust these values in such a way that the difference between the 2 values becomes larger than a certain threshold value. We name this threshold value as “Watermark Strength Parameter” because this value decides the robustness of watermark data. Certainly, it has an impact on the image perceptibly. So, we have to decide this threshold value in such a way that our image does not lose its quality much.

Out of these 7 tuples, last 3 tuples are algorithms which are discussed below:

**5.3.1 G, THE POLICY GENERATOR ALGORITHM**

Similar to our earlier proposed ICAR watermarking scheme for the gray image watermarking, we had to watermark each copy $X_i$ of an image $X$ differently. Therefore, we need a different watermarking policy for each copy of the image to be watermarked. Here “Policy” means that for every copy of the image, there will be unique combination of 4 middle band coefficients. To generate a policy, we simply take $8 \times 8$ DCT of the input image $X_i$ and randomly select 4 coefficients out of 22 middle band coefficient of FM region from any of the red, green or blue color channel. So, numbers of policies that can be generated are $^{22}C_4 = 7315$ which means that 7315 copies of a single image can be watermarked such that no two watermarked images have same policy. This step ensures that attacker can not conclude the location of watermark data by colluding many
watermarked copies of an image. This also depicts that our proposed scheme is an ICAR scheme.

5.3.2 COLOR CHANNEL SELECTION
Up to the development of this approach, we used only “BLUE” color channel to hide the watermark data. Bossen et al. [9] have stated that the watermarks should be embedded mainly in the BLUE color channel of an image. The human eye is least sensitive to change in BLUE channel. However, the suitability of color channel to hide the watermark data is dependent on the image itself and therefore, we have discussed some interesting results related to this issue in the Chapter 6. In this section, we are using BLUE channel to hide the watermark data.

5.3.3 E, THE WATERMARK EMBEDDING ALGORITHM
In this algorithm, each 8x8 DCT block of an image is used to hide a single bit of watermark logo. Our embedding algorithm is based on averaging the coefficients of FM region. We can fight against collusion attack by swapping more than one pair but if attacker is ready to loose some quality, he/she can disturb all the coefficients in FM region. Therefore, even if we introduce redundancy with randomness, our watermark data may still be attacked. So, we propose that an attacker cannot alter the image such that the “average” of coefficients of FM region changes much. Accordingly, we are hiding “1” or “0” by using relative value of a coefficient and the average “Av” of coefficients of FM region. This algorithm is given as below:

1. Repeat steps 2 to 11 for i = 1…..n;

   // where 'n' is the number of copies of a single image to be watermarked //

2. INPUT (X_i);
3. Take 8x8 block DCT of X_i;
4. INPUT (W);
5. Convert W into a string S = (S_j | S_j = {0,1}, for j = 1…..length of the watermark);
6. Let \( L = \text{STRING\_LENGTH}(S); \)
   // \( L \) is the length of watermark data. If \( L = 1000 \), then first 1000 DCT block of \( X_i \) are used //

7. \( P_i = \text{CALL}(G); \)
   // Each generated \( P_i \) shall be stored in an author’s database for the detection purpose in future.
   Let the \( P_i \) for chosen \( X_i \) be, \( P_i = \{ (5,1), (4,2), (6,3) \text{ and } (5,4) \} \) in BLUE channel //

8. Calculate the average “Av” of remaining 18 middle band coefficients.

9. Repeat steps 10 to 11 for \( r = 1 \ldots L; \)

10. Read \( S_r; \)

   // Now like classical MBCE scheme, relative strength of average “Av” and chosen 4 coefficients in step 7 will interpret “0” or “1” of watermark data. To hide “0”, for all 4 chosen coefficients in step 7, we assigned the value of coefficients which is ‘\( T \)’ less than the average “Av”. To hide “1”, for all 4 chosen coefficients in step 7, we assigned the value of coefficients which is ‘\( T \)’ greater than the average “Av” //

   If \((S_r = 0)\)
   \[
   \begin{align*}
   \text{DCT}(5,1) &= \text{Av} - T; \\
   \text{DCT}(4,2) &= \text{Av} - T; \\
   \text{DCT}(5,4) &= \text{Av} - T; \\
   \text{DCT}(6,3) &= \text{Av} - T;
   \end{align*}
   \]

   Else
   \[
   \begin{align*}
   \text{DCT}(5,1) &= \text{Av} + T; \\
   \text{DCT}(4,2) &= \text{Av} + T; \\
   \text{DCT}(5,4) &= \text{Av} + T; \\
   \text{DCT}(6,3) &= \text{Av} + T;
   \end{align*}
   \]

   End;

11. Take IDCT to reconstruct \( X_i; \)

12. End.

5.3.4 D, THE WATERMARK DETECTION ALGORITHM

Watermark extraction is the reverse procedure of watermark embedding. To extract the watermark from the watermarked image, we calculated the average “Av” in the same way as in embedding algorithm. Owner should have a record of all policies used to watermark
the image. Based on “policies”, owner of the image can recover watermark using following rules:

1) If at least 1 out of 4 chosen coefficients are less than average, Interpret “0”; and
2) If at least 1 out of 4 chosen coefficients are greater than average, interpret “1”.

The detection algorithm steps are as follows:

1. INPUT (X\_i\');
   
   // Xi’ is the attacked copy of a watermarked image //

2. Take 8x8 block DCT of X\_i’ and calculate Av;

3. For all P\_i in author’s database, repeat the steps 4;
   
   // If initially 10 copies were watermarked, then out of 10 policies, for 1 policy, watermark will be
   // recovered correctly. To explain further steps, we are assuming that now algorithm is in a loop
   // where Pi is (5,1) (4,2) (5,4) and (6,3), which was used to watermarked this particular Xi’ //

4. Repeat the steps 5 for j = 1….L;
   
   // L is the length of watermark data. A single bit will be recovered form one 8x8 DCT block//

5. Take j\textsuperscript{th} DCT block to form j\textsuperscript{th} bit of watermark as follows:

   If (DCT (5, 1) <= Av)
   
   T1 = 1;

   Else T1 = 0;

   If (DCT (4, 2) <= Av)
   
   T2 = 1;

   Else T2 = 0;

   If (DCT (5, 4) <= Av)
   
   T3 = 1;

   Else T3 = 0;

   If (DCT (6, 3) <= Av)
   
   T4 = 1;

   Else T4 = 0;

   If (T1 + T2 + T3 + T4 >= 1)
Decode “0”
If (DCT (5, 1) > Av)
    P1 = 1;
Else  P1 = 0;
If (DCT (4, 2) > Av)
    P2 = 1;
Else  P2 = 0;
If (DCT (5, 4) > Av)
    P3 = 1;
Else  P3 = 0;
If (DCT (6, 3) > Av)
    P4 = 1;
Else  P4 = 0;
If ( P1 + P2 + P3 + P4  >=  1)
    Decode “1”
End;

6. Store W’, the recovered watermark;
7. End.

It may be observed from both the algorithms that even if attacker alters the values of the coefficient of FM region, if “Av” is not changed much, then we can recover the watermark data correctly and attacker cannot aim to attack the image in such a manner which modifies “Av”.

5.3.5 PERFORMANCE OF THE PROPOSED SCHEME
Our proposed scheme does not need any testing to check whether or not it is robust against the collusion attack, as it is designed in such a way that the attacker cannot analyze the pattern by colluding many watermarked copies. We needed to check the performance of the proposed scheme against the JPEG compression and other common image manipulations and known attacks. For this, we tested our scheme on 3 test images
Firstly, we chose an appropriate value of “T” which affects least the image quality as well as optimizes the recovery of watermark data. Our experiments suggested that if we were hiding the watermark using $T = 150$, there was approximately no loss in the perceptual quality of the images and recovered watermark logos were of very fine quality. Figure 5.2 shows the watermarked test images after hiding watermark logo by keeping $T = 150$. It may be seen that, images are not disturbed at all. Figure 5.3 shows the extracted watermark logos from these watermarked copies of Lena, Mandrill and Pepper without performing any attack or manipulations on the watermarked images. This fixed up the value of $T = 150$ for further tests.

![Figure 5.2: Watermarked test images keeping $T = 150$](image1)

![Figure 5.3: Extracted watermark from watermarked Lena, Mandrill and Pepper images respectively at $T = 150$](image2)

**5.3.5.1 PERFORMANCE AGAINST JPEG COMPRESSION:** We applied JPEG compression on watermarked images (generated by keeping $T = 150$) at different JPEG quality parameters $Q$ and then recovered the watermark logos. Table 5.3 summarizes the
PSNR of extracted watermark logos. It may be observed from Table 5.3 that even at $Q = 20$, quality of extracted watermark is very fine and logos are quite detectible.

### 5.3.5.2 PERFORMANCE AGAINST COMMON IMAGE MANIPULATIONS:

We performed the following attacks on the watermarked test images:

- **Attack-1**: Equalize the Histogram;
- **Attack-2**: Apply uniform scaling (Zoom);
- **Attack-3**: Adjust the brightness to +40 and contrast to +25;
- **Attack-4**: Adjust the hue and saturation to +10 each;
- **Attack-5**: Add 10% Gaussian noise; and
- **Attack-6**: Blur the image using Gaussian blur with 1 pixel radius.

<table>
<thead>
<tr>
<th>Quality factor</th>
<th>Lena Watermarked with $T = 150$</th>
<th>Mandrill Watermarked with $T = 150$</th>
<th>Pepper Watermarked with $T = 150$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q = 80$</td>
<td>39.9987</td>
<td>37.0185</td>
<td>39.9987</td>
</tr>
<tr>
<td>$Q = 60$</td>
<td>39.9987</td>
<td>34.98135</td>
<td>39.9987</td>
</tr>
<tr>
<td>$Q = 40$</td>
<td>24.57225</td>
<td>14.51025</td>
<td>25.20285</td>
</tr>
<tr>
<td>$Q = 20$</td>
<td>21.92385</td>
<td>12.26715</td>
<td>21.3678</td>
</tr>
</tbody>
</table>

Then, we recovered the watermark logos from attacked images and calculated the PSNR value of watermark logos. Table 5.4 summarizes the PSNR values of extracted logos recovered from all test images. Our proposed scheme sustained all the attacks and the quality of the extracted watermark logos is quite good. Figure 5.4 shows the recovered logos from attacked images.
Table 5.4: PSNR of extracted watermark logo from watermarked test images after attacks

<table>
<thead>
<tr>
<th>PSNR (DB)</th>
<th>Lena</th>
<th>Mandrill</th>
<th>Pepper</th>
</tr>
</thead>
<tbody>
<tr>
<td>Histogram Equalization</td>
<td>34.67</td>
<td>28.06</td>
<td>32.25</td>
</tr>
<tr>
<td>Zoom</td>
<td>34.67</td>
<td>28.04</td>
<td>32.07</td>
</tr>
<tr>
<td>Brightness-Contrast-Adjustment</td>
<td>34.67</td>
<td>28.04</td>
<td>30.78</td>
</tr>
<tr>
<td>Hue-Saturation</td>
<td>34.67</td>
<td>28.04</td>
<td>32.48</td>
</tr>
<tr>
<td>Gaussian Noise</td>
<td>34.67</td>
<td>28.04</td>
<td>31.78</td>
</tr>
<tr>
<td>Gaussian Blur</td>
<td>34.67</td>
<td>28.04</td>
<td>31.10</td>
</tr>
</tbody>
</table>

5.3.5.3 COMPARATIVE STUDY RESULTS WITH OTHER SCHEMES: We compared our scheme against JPEG compression with other similar and state-of-the-art methodologies which are well known for their robustness against JPEG compressions. The chosen schemes are as follows:

**Scheme-A:** Correlation based Schemes with 2 PN sequence (Section 2.1.3.2)

**Scheme-B:** Classical MBCE Scheme (Section 2.2.2.1)

**Scheme-C:** Scheme proposed in Section 4.4 is also based on Middle Band Coefficient Exchange (MBCE) scheme and ICAR in nature. So, we decided to compare the performance of our scheme with this scheme also. This scheme swaps 4 pairs of coefficients in FM region in correlation with low band coefficients. We are naming this scheme as Scheme-C.

Then, we re-implemented the chosen schemes for the colored images and hid the watermark data in BLUE channel.

**Scheme-D:** We are naming our proposed scheme as Scheme-D.

It is observed that all the above schemes are robust against JPEG compression attack but if we compress the watermark images by very low quality factors (less than $Q = 20$), our proposed scheme outperforms the other schemes. We compressed the watermarked test
images by keeping JPEG quality factor Q = 15, 10, and 5. No scheme, other than the proposed one, was able to extract the detectible watermark logos.

Table 5.5 summarizes the PSNR values of extracted logos from highly compressed watermark test images using various schemes. Figure 5.5 shows the recovered watermark logos from highly compressed watermarked images using our proposed scheme. It may be observed that recovered logos are quite detectible and proposed scheme is more efficient than the other chosen schemes.
Table 5.5: PSNR values of extracted logos from highly compressed watermarked test images using various schemes

<table>
<thead>
<tr>
<th>Schemes</th>
<th>JPEG Quality Factors</th>
<th>Lena</th>
<th>Mandrill</th>
<th>Pepper</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scheme-A</td>
<td>Q = 15</td>
<td>8.723</td>
<td>7.89</td>
<td>8.12</td>
</tr>
<tr>
<td></td>
<td>Q = 10</td>
<td>7.67</td>
<td>7.12</td>
<td>7.988</td>
</tr>
<tr>
<td></td>
<td>Q = 05</td>
<td>4.5</td>
<td>4.324</td>
<td>4.657</td>
</tr>
<tr>
<td>Scheme-B</td>
<td>Q = 15</td>
<td>4.222</td>
<td>4.587</td>
<td>3.987</td>
</tr>
<tr>
<td></td>
<td>Q = 10</td>
<td>3.45</td>
<td>3.87</td>
<td>3.95</td>
</tr>
<tr>
<td></td>
<td>Q = 05</td>
<td>2.32</td>
<td>2.2</td>
<td>1.97</td>
</tr>
<tr>
<td>Scheme-C</td>
<td>Q = 15</td>
<td>4.323</td>
<td>4.565</td>
<td>4.33</td>
</tr>
<tr>
<td></td>
<td>Q = 10</td>
<td>4.11</td>
<td>4.249</td>
<td>4.12</td>
</tr>
<tr>
<td></td>
<td>Q = 05</td>
<td>2.234</td>
<td>2.229</td>
<td>2.1</td>
</tr>
<tr>
<td>Scheme-D</td>
<td>Q = 15</td>
<td>16.305</td>
<td>10.845</td>
<td>13.335</td>
</tr>
<tr>
<td></td>
<td>Q = 10</td>
<td>15.585</td>
<td>10.62</td>
<td>12.885</td>
</tr>
<tr>
<td></td>
<td>Q = 05</td>
<td>14.13</td>
<td>10.29</td>
<td>11.4</td>
</tr>
</tbody>
</table>

Test Images / JPEG Q Factor

<table>
<thead>
<tr>
<th>Lena</th>
<th>Mandrill</th>
<th>Pepper</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q = 15</td>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
</tr>
<tr>
<td>Q = 10</td>
<td><img src="image4.png" alt="Image" /></td>
<td><img src="image5.png" alt="Image" /></td>
</tr>
<tr>
<td>Q = 05</td>
<td><img src="image7.png" alt="Image" /></td>
<td><img src="image8.png" alt="Image" /></td>
</tr>
</tbody>
</table>

Figure 5.5: Extracted logos using proposed scheme from highly compressed watermarked test images
Results indicate that proposed scheme recovers the watermark even from an attacked image which is compressed up to Q = 5 quality factor of JPEG (i.e. after 95 - 99% size reduction).

This proves that the proposed scheme is not only an ICAR scheme but also very robust to JPEG compression. In addition to this, the proposed scheme is resisting common image manipulations like cropping, scaling, flipping, histogram equalization, brightness-contrast adjustment, Hue-saturation alteration, Gaussian noise and Gaussian blur.

5.4 CONCLUSION

In this chapter, we have discussed the watermarking of the colored images. Since a colored image has R, G and B color channel, firstly we presented a study to find the suitability of a color channel to carry the watermark data with respect to the robustness against an attack. It was found that if an image has to undergo JPEG compression attack, then the watermark data should be hidden in GREEN color channel to ensure the best recovery of the watermark logo. Then, we presented an ICAR watermarking scheme based on the “average” of the FM coefficients. Results indicted that the proposed scheme is very robust against JPEG compression and common image manipulations and better then other similar state-of-the-art schemes.